CITY UNIVERSITY
LONDON

Assurance Cases in supply change risk management: opportunities and threats

DHS SW Assurance Forum, Sept 2010

Robin E Bloomfield
Adelard LLP and CSR City University London

reb@adelard.com
reb@csr.city.sc.uk
College Building, City University, London EC1V 0HB
Tel: +44 20 7490 9450 (sec Adelard)
Tel: +44 20 7040 8420 (sec CSR)

CSR Building confidence in a computerised world
www.csr.city.ac.uk

Adelard

Overview

- Introduction
- Safety and assurance practices
- Supply chain experience
  - nuclear smart devices
  - financial system
- Extending to SCRM
- Threats and opportunities
- Conclusions and discussions

Adelard

- Safety and assurance cases and safety management systems
- Independent safety assessment
- Software assurance, including formal methods and static analysis
- Development, interpretation and application of standards and guidelines
- applied research in safety, security, critical infrastructure interdependencies
- policy to technology
- ASCE – the Assurance and Safety Case Environment
- clients in nuclear, defence, financial, transport sectors

Centre for Software Reliability

- Evaluation of socio-technical systems
  - Technical, interdisciplinary
- Research
  - with international community and users
- Education
  - placements, internships, scholarships, courses, MSc and CPD
- Innovation
  - director, Dr Peter Popov
  - DivSQL, PIA-FARA

In the beginning…

- “The World, according to the best geographers, is divided into Europe, Asia, Africa, America, and Romney Marsh",

wrote the Reverend Richard Harris Barham, writing as Thomas Ingoldsby, in the 1840s.

Some Definitions

“A documented body of evidence that provides a
convincing and valid argument that
adequately safe for
env

A structured ***argument***, supported by a body of ***evidence***, that provides a compelling, comprehensible and valid case that a ***system is safe*** for a given application in a given
ent

A security assurance case us
and a correspond
system s
properti

A f
and
that
railwa
the sa

ts
g assurance
er change to the
afety requirements and that
quirements are adequate.

Yellow Book issue 4

An assurance case is reasoned, auditable artefact created to support the contention its claim or claims are satisfied. It contains the following and their relationships:

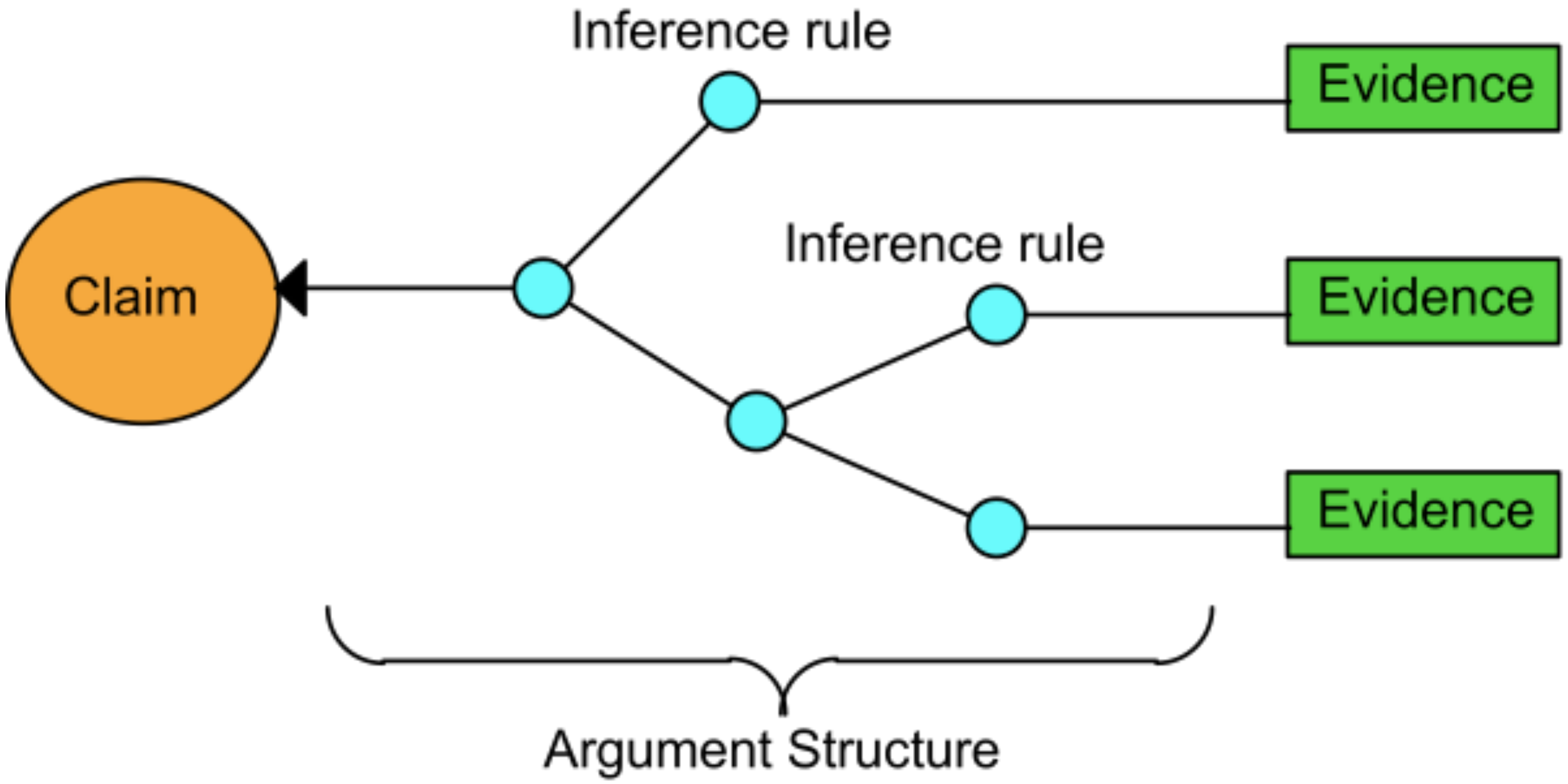
One or more claims about properties.

Arguments that logically link the evidence and any assumptions to the claim(s).

A body of evidence and possibly assumptions supporting these arguments for the claim(s).

ISO 15026

Safety cases



- “a documented body of evidence that provides a convincing and valid argument that a system is adequately safe for a given application in a given environment”

Elements of a “Case”

- Claim about a property of the system or some subsystem, with some confidence.
- Evidence that used as the basis of the trust argument. This can be either facts (e.g. based on established scientific principles and prior research), assumptions, or sub-claims, derived from a lower-level sub-argument.
- Argument linking the evidence to the claim, which can be deterministic, probabilistic or qualitative.

Types of argument

Deterministic or analytical application of predetermined rules to derive a true/false claim (given some initial assumptions), e.g. formal proof (compliance to specification, safety property), execution time analysis, exhaustive test, single fault criterion

Probabilistic quantitative statistical reasoning, to establish a numerical level, e.g. MTTF, MTTR, reliability testing

Qualitative compliance with rules that may have an indirect link the desired attributes, e.g. compliance with QMS and safety standards, staff skills and experience

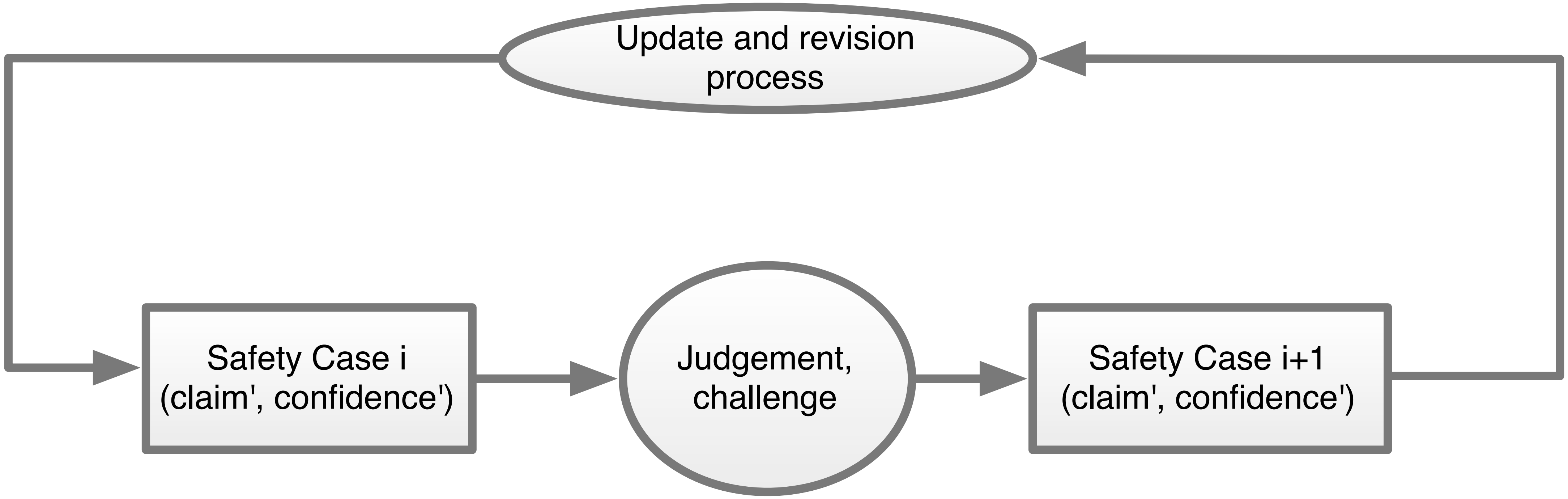
Making arguments explicit a key idea
Separating evidence from information

Communication and reasoning

- Structured safety and assurance cases have two essential roles:
  - communication is an essential function of the case, from this we can build confidence
    - boundary objects that record the shared understanding between the different stakeholders
  - a method for reasoning about dependability (safety, security, reliability, resilience ...) properties of the system
- Both are required to have systems that are trusted and trustworthy

Safety case process – building confidence, challenging assumptions

- Captured in safety management system and in meta-case
- Challenge and response cycle essential
- Proof as a social, technical, adversarial process

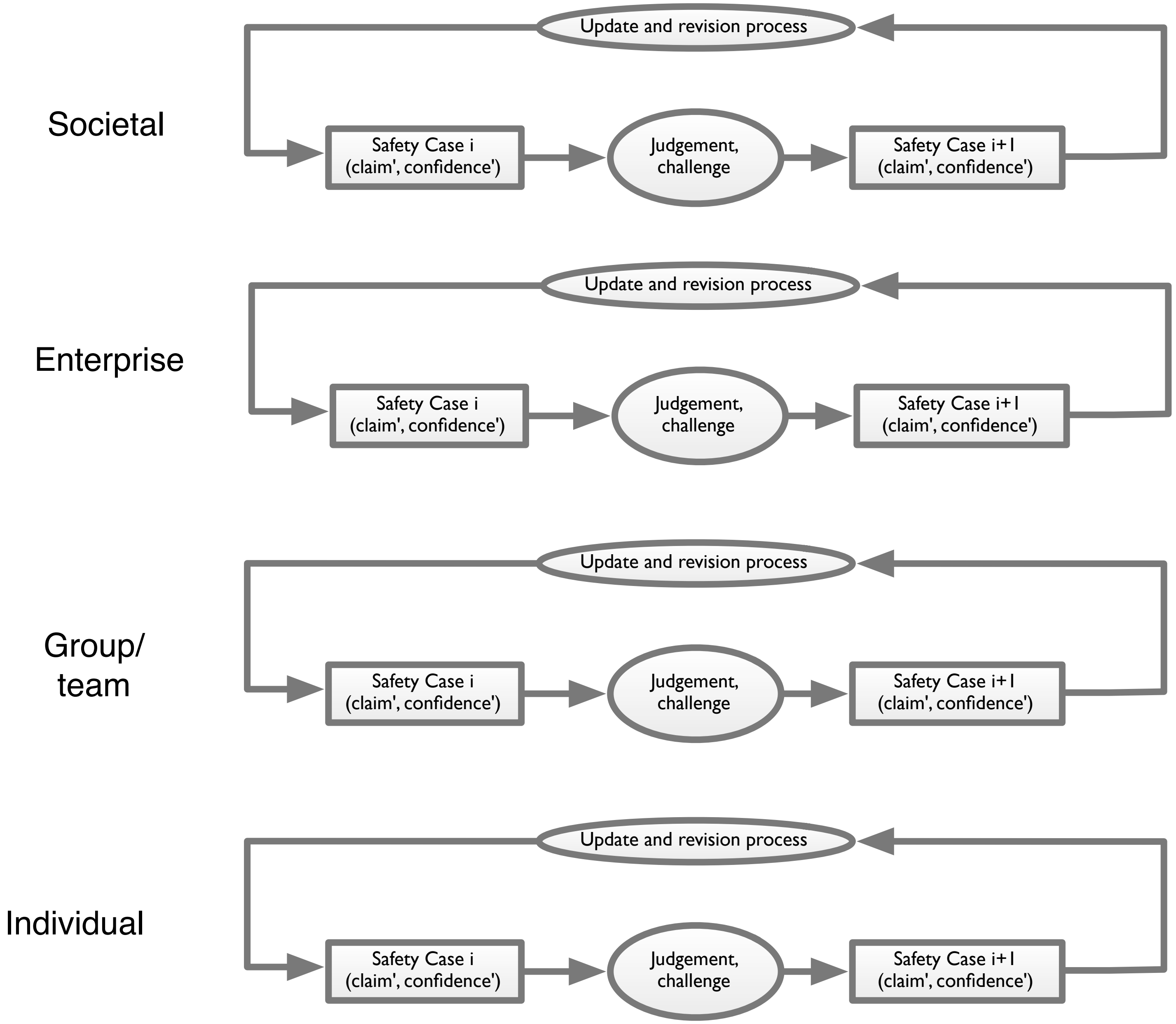

Update and revision process
Safety Case i (claim', confidence')
Judgement, challenge
Safety Case i+1 (claim', confidence')

CSR Building confidence in a computerised world
www.csr.city.ac.uk

Adelard

Safety case process – building confidence, challenging assumptions

- Captured in safety management system and in meta-case
- Challenge and response cycle essential
- Proof as a social, technical, adversarial process



Societal

Update and revision process

Safety Case i (claim', confidence')

Judgement, challenge

Safety Case i+1 (claim', confidence')

Enterprise

Update and revision process

Safety Case i (claim', confidence')

Judgement, challenge

Safety Case i+1 (claim', confidence')

Group/ team

Update and revision process

Safety Case i (claim', confidence')

Judgement, challenge

Safety Case i+1 (claim', confidence')

Individual

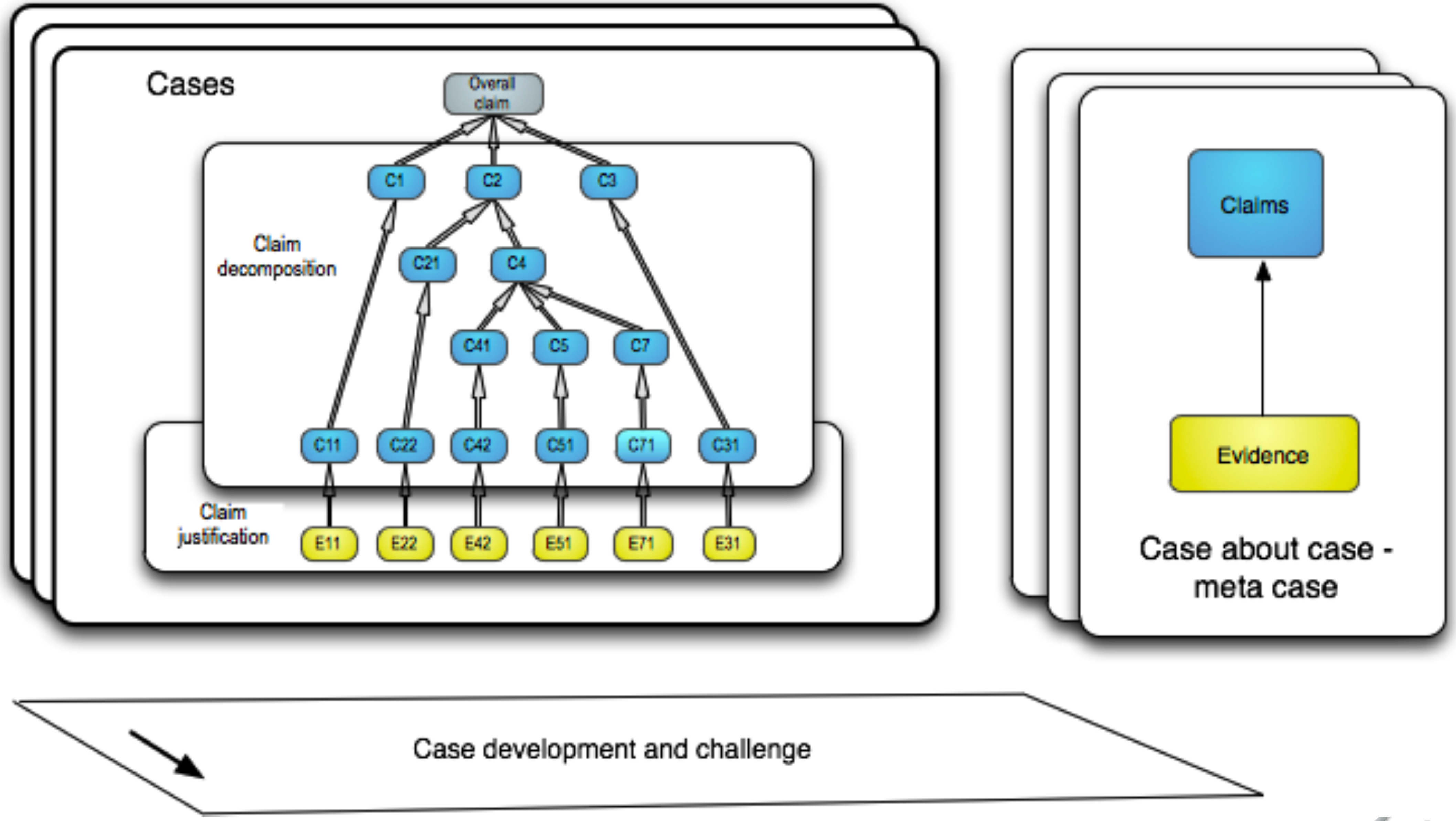
Update and revision process

Safety Case i (claim', confidence')

Judgement, challenge

Safety Case i+1 (claim', confidence')

Reasoning, communication, confidence

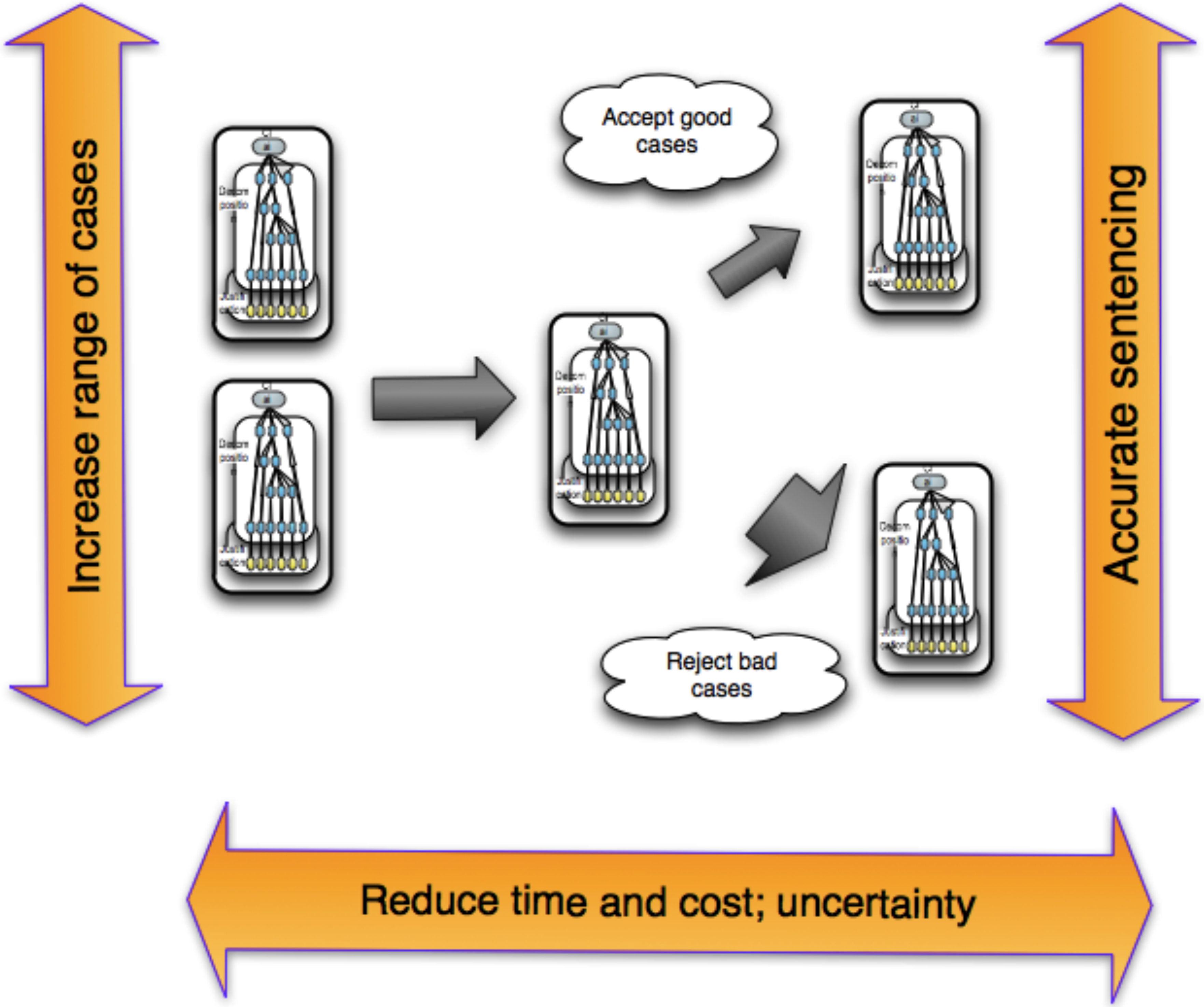


CSR Building confidence in a computerised world

www.csr.city.ac.uk

Adelard

Objectives



Increase range of cases

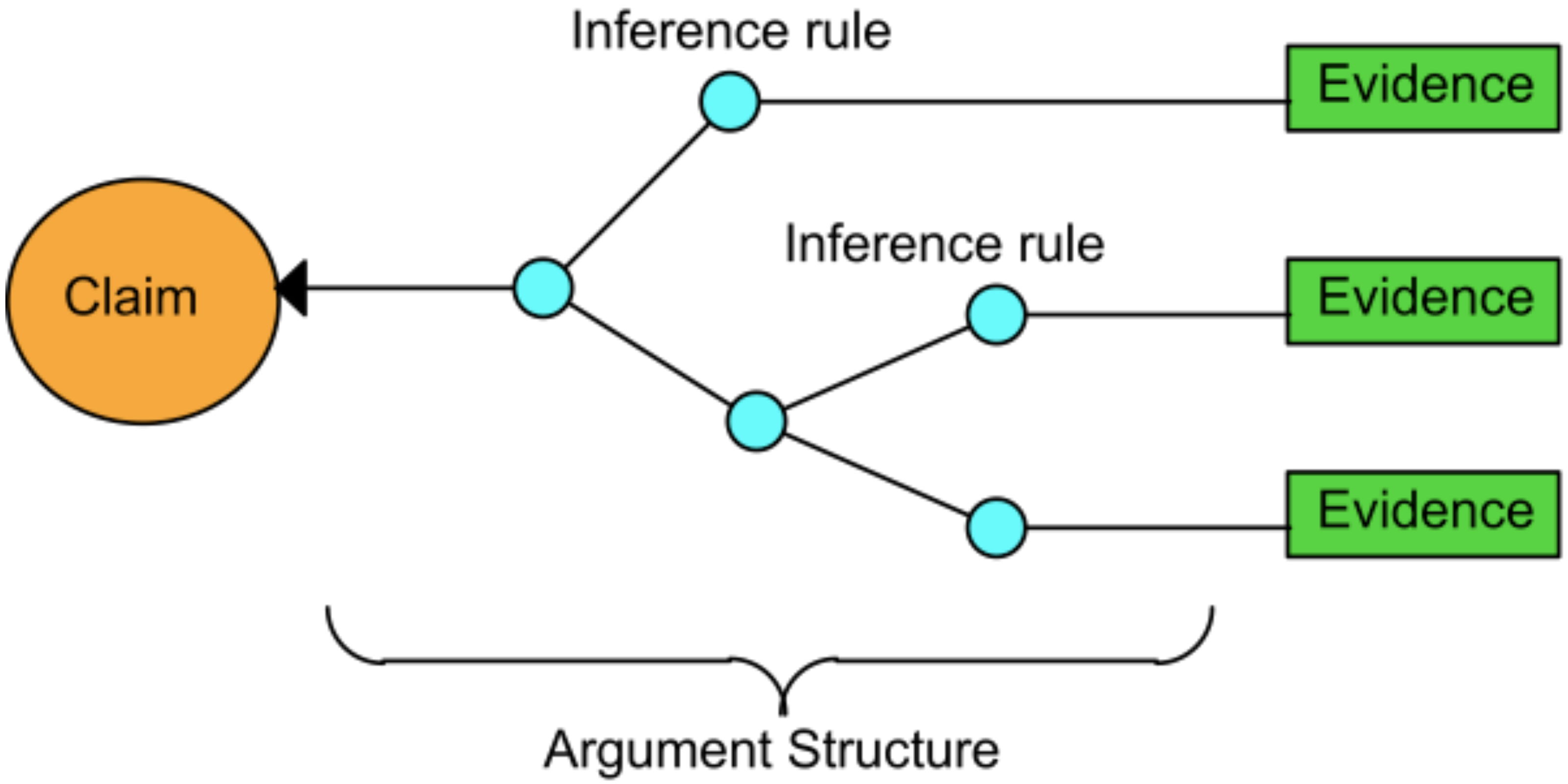
Accept good cases

Reject bad cases

Accurate sentencing

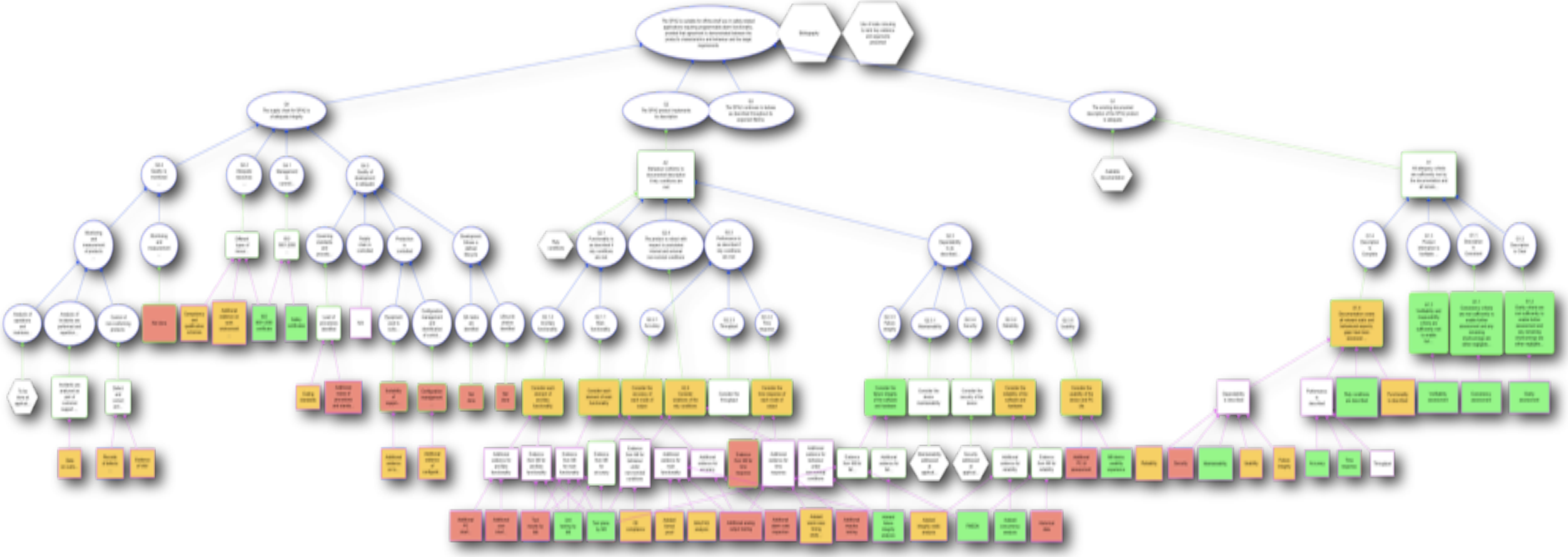
Reduce time and cost; uncertainty

In theory ...



- "a documented body of evidence that provides a convincing and valid argument that a system is adequately safe for a given application in a given environment"

In practice …

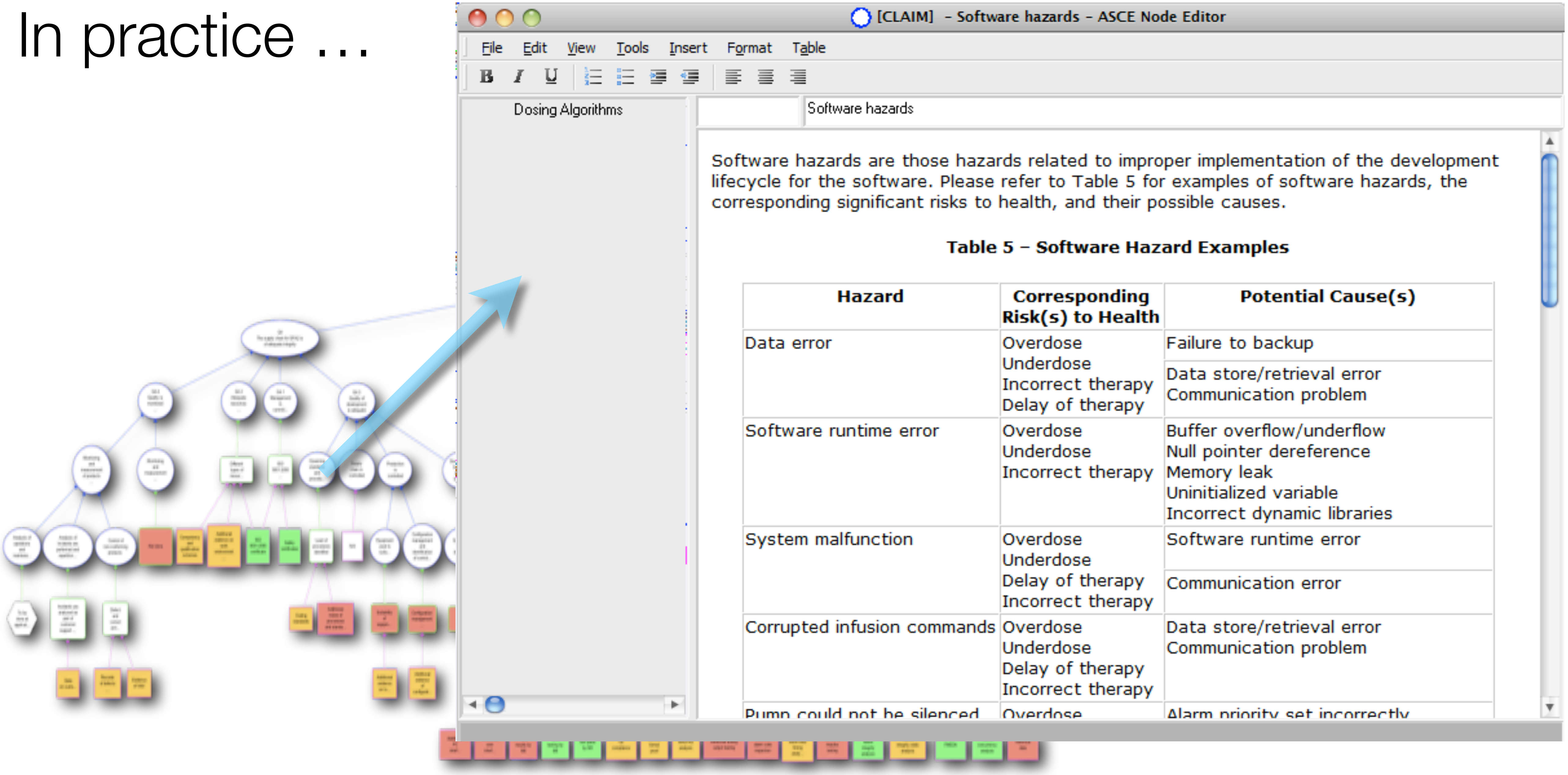


CSR Building confidence in a computerised world

www.csr.city.ac.uk

Adelard

In practice …



Software hazards are those hazards related to improper implementation of the development lifecycle for the software. Please refer to Table 5 for examples of software hazards, the corresponding significant risks to health, and their possible causes.

Table 5 – Software Hazard Examples

Hazard	Corresponding Risk(s) to Health	Potential Cause(s)
Data error	Overdose Underdose Incorrect therapy Delay of therapy	Failure to backup Data store/retrieval error Communication problem
Software runtime error	Overdose Underdose Incorrect therapy	Buffer overflow/underflow Null pointer dereference Memory leak Uninitialized variable Incorrect dynamic libraries
System malfunction	Overdose Underdose Delay of therapy Incorrect therapy	Software runtime error Communication error
Corrupted infusion commands	Overdose Underdose Delay of therapy Incorrect therapy	Data store/retrieval error Communication problem
Pump could not be silenced	Overdose	Alarm priority set incorrectly

Architecting claim structure

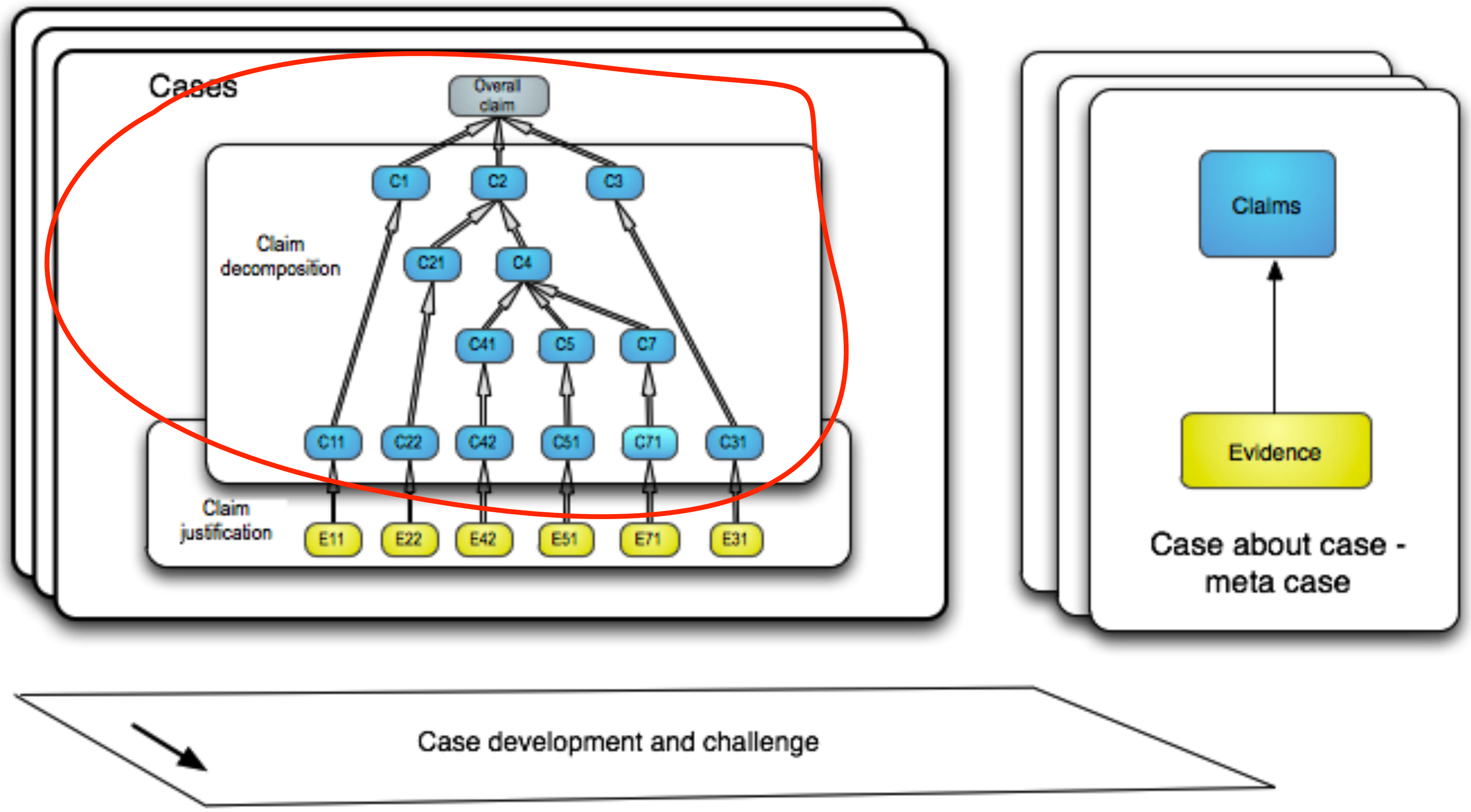
CSR Building confidence in a computerised world

www.csr.city.ac.uk

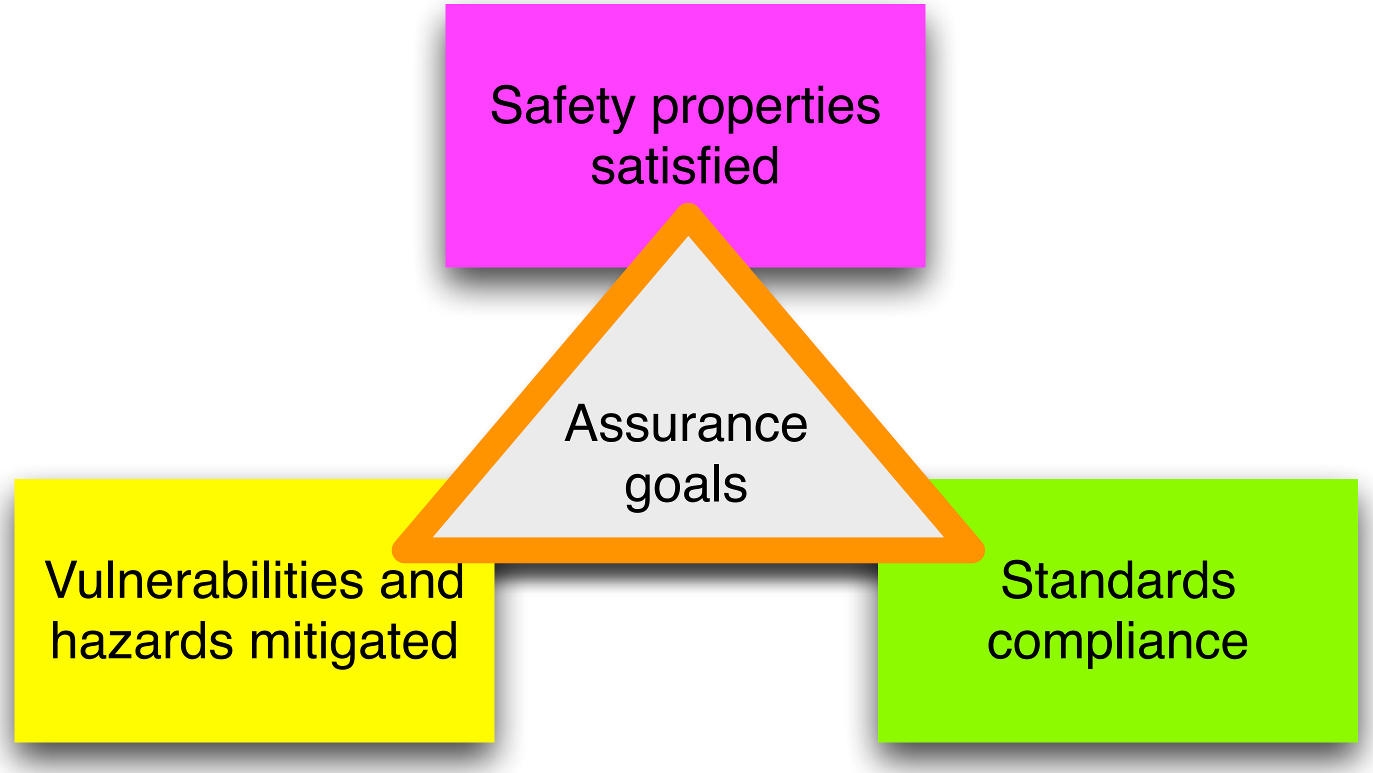
Adelard

Claim structure

- creative strategies
- claims language
- templates



Approaches


Safety properties satisfied
Assurance goals
Vulnerabilities and hazards mitigated
Standards compliance

Cases - argument styles

We have done what we were told to do (a *standards compliance* argument)

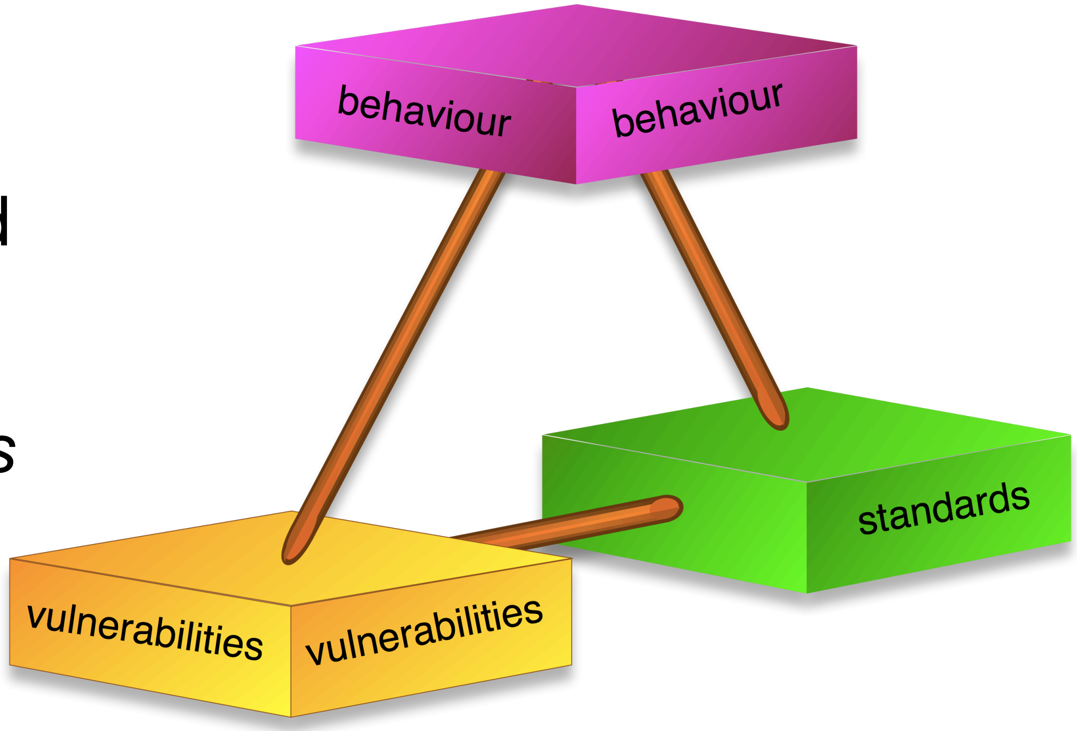
The system achieves the behaviour required (*safety properties* satisfied)

The system does not do bad things (*hazards addressed, vulnerabilities mitigated*)

Also

We have tried very hard (a *process argument*) to achieve dependability

Often a mixture of styles will be incorporated into a single case.



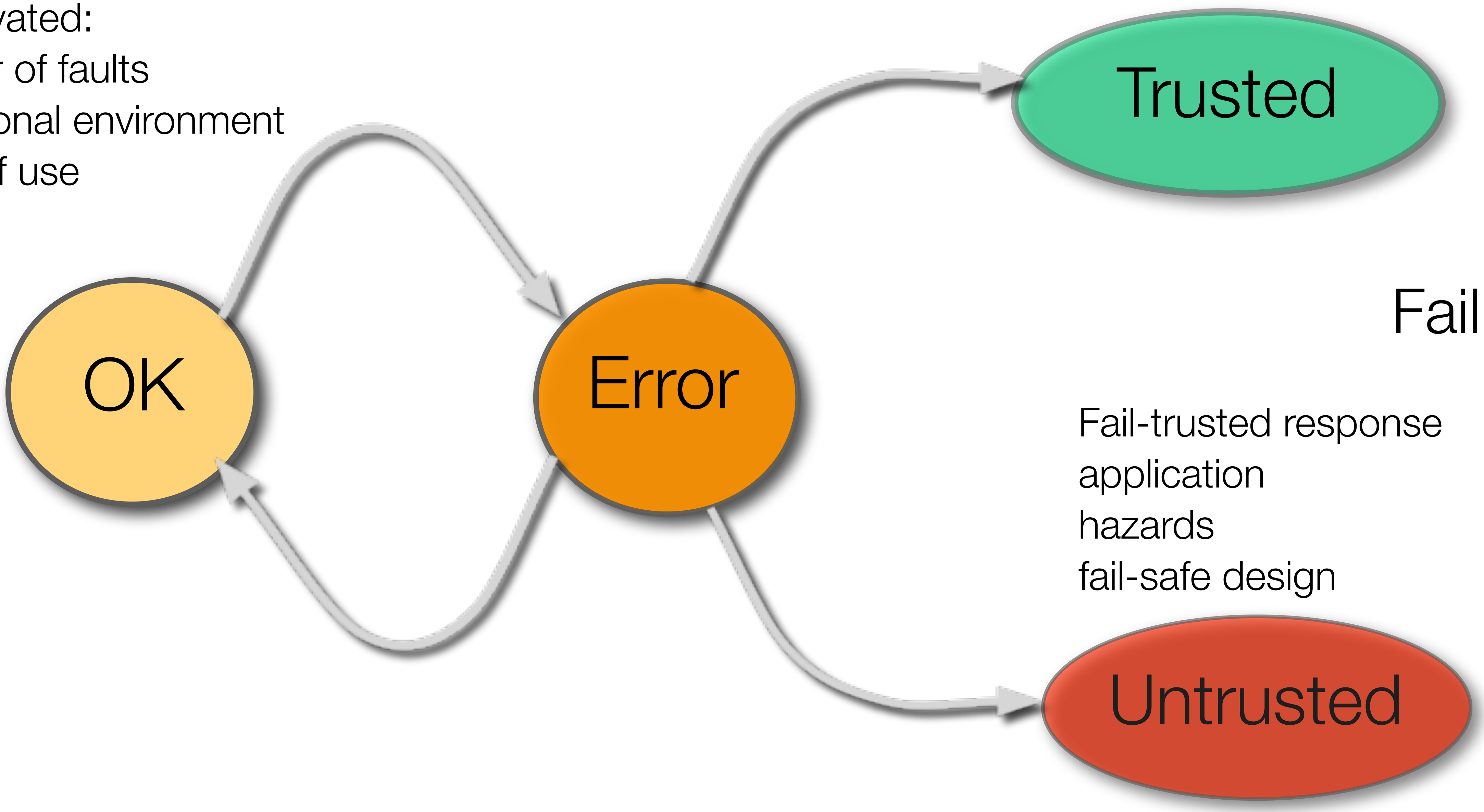
Standards and regulations

- Important part of case
- Can play different roles
  - Which needs to be justified
- But issues of validation
  - process -> product
  - techniques -> SIL achieved
- Need to innovate
  - Technology development V&V moves on
  - Use of COTS products
  - Product lines
  - Compliance can be expensive

Assurance strategies - behaviour

Fault activated:

- Number of faults
- Operational environment
- Mode of use



Failures

Fail-trusted response
application
hazards
fail-safe design

fault tolerance in design nature of application --
self healing, grace time

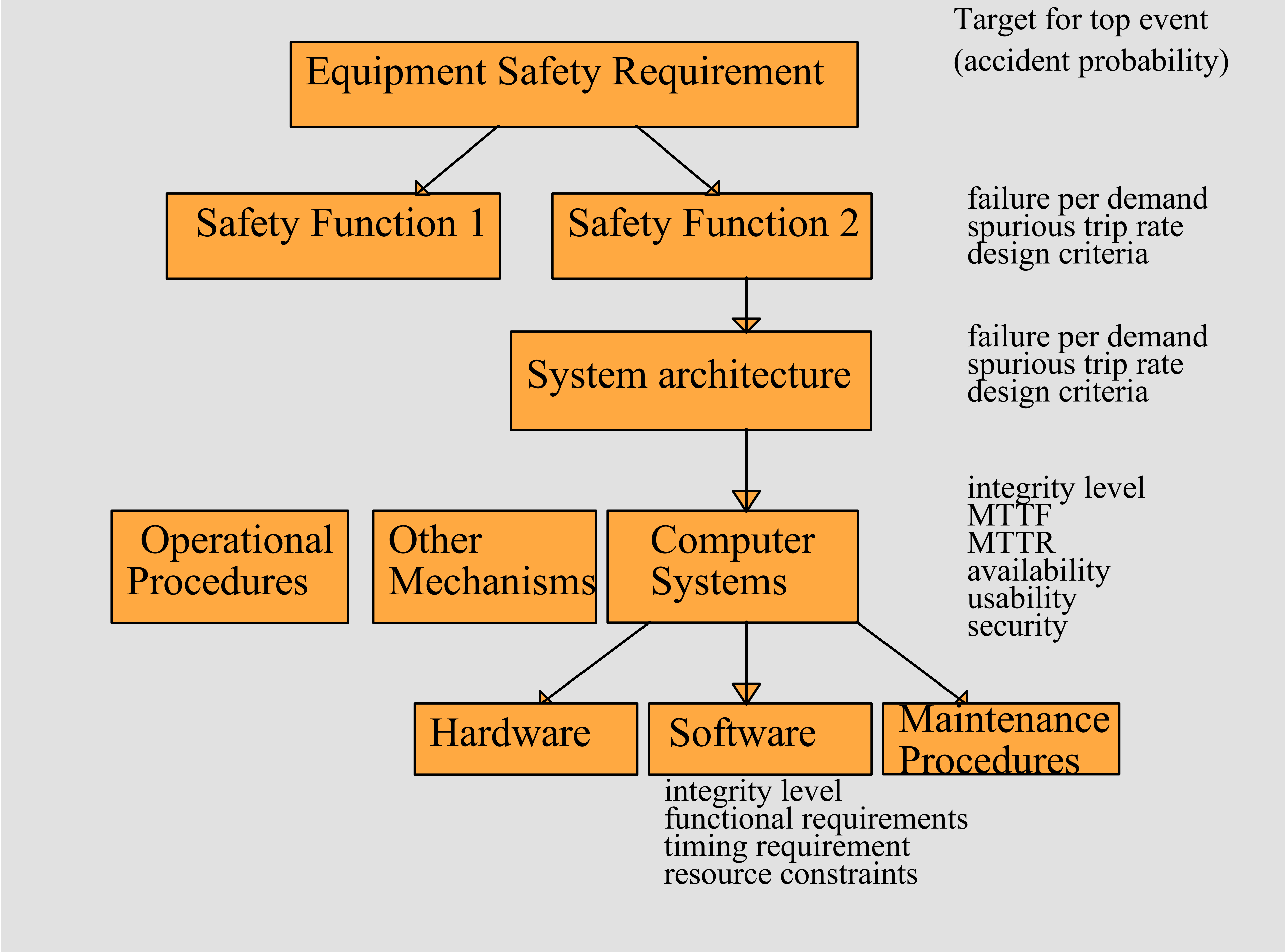
Strategies on behaviour

- Strategy – N No critical/significant fault or unsafe feature exists (the beast has no teeth, claws)
- Strategy –W Wrapper/containment argument – no failure or feature of the component can lead to hazard (the beast is in the cage)
- Strategy –R Restoration argument – any failure can be detected and recovered from (the beast can always be put back in the cage)
- And probabilistic variants of these

Safety properties and claims

- System safety analysis identifies hazards; these are amalgamated and abstracted into safety properties.
- Safety properties can be functions (shut down when T> 500), invariants (min sep always >2 miles) or purely descriptive (competency and culture).
- For each safety property address all attributes to increase completeness.
- As the design progresses need to consider derived properties arising from hazards introduced by the implementation.
- Non-functional system properties evolve
- May be claim limits

Architecture and functional claim expansion



Target for top event
(accident probability)

failure per demand
spurious trip rate
design criteria

failure per demand
spurious trip rate
design criteria

integrity level
MTTF
MTTR
availability
usability
security

integrity level
functional requirements
timing requirement
resource constraints

Claim attribute expansion

- Claims can be broken down into claims about different attributes for the various sub-systems, e.g.:

reliability and availability
usability (by the operator)
security (external attack)
fail-safe response
functional correctness

accuracy
time response
robustness to overload
maintainability
modifiability, etc.

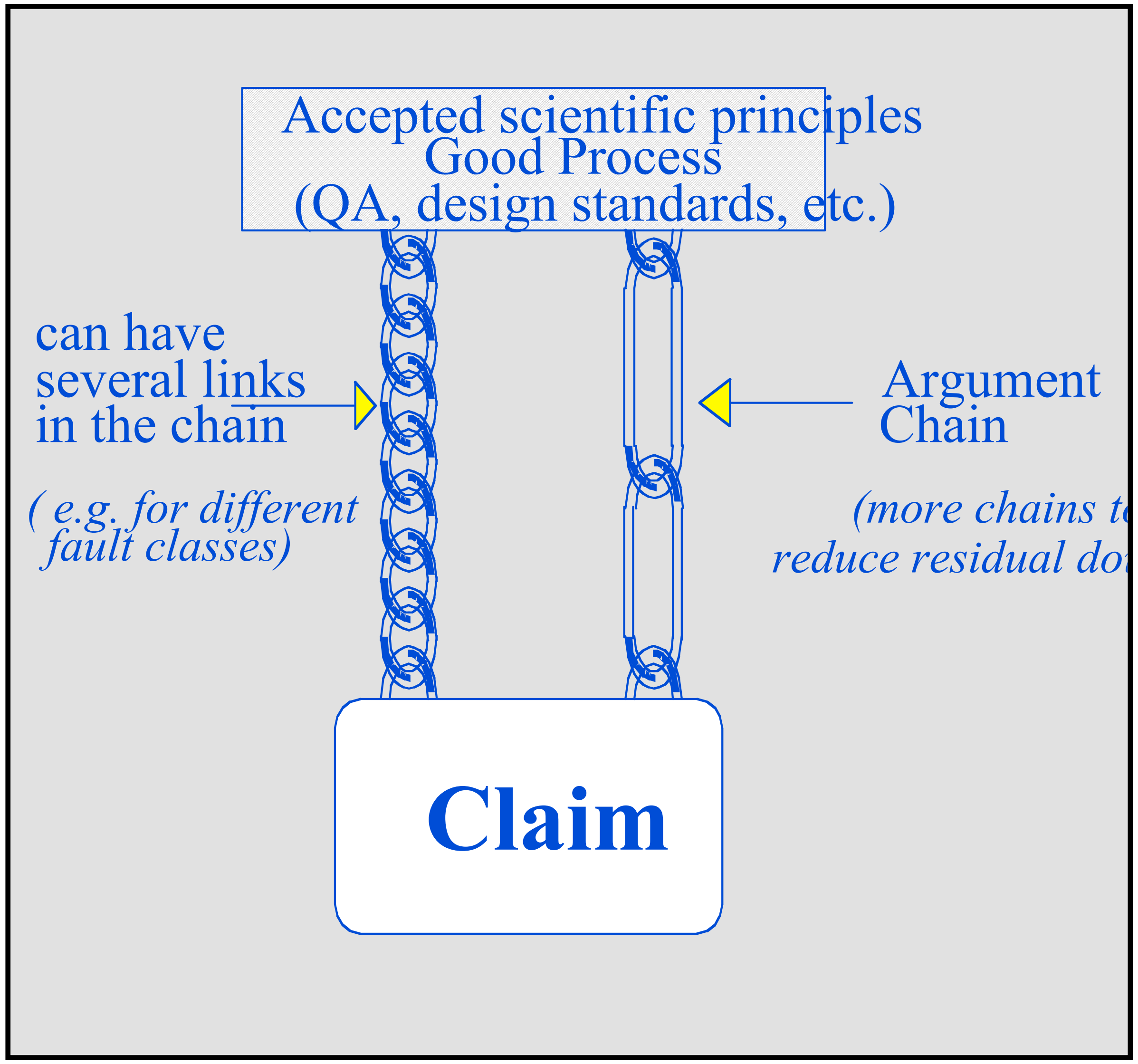
Restricted types of claim expansion

- Claim expansion language initially unconstrained
  - CAE
  - (also of course GSN)
- Empirically found a small set of constructs useful
- These enable more formal underpinnings and pragmatic checklists
- Uniformity and regularity in cases
- Allows us to asses cases
- Gradually introduced in our work

Main types – keywords	Comment
architecture	splitting a component into several others
functional	
property decomposition	splitting a property into several others e.g. set of attributes
infinite set	inductive partitioning (e.g., over time)
complete	capturing the full set of values for risks, requirements, etc.
monotonic	the new system only improves on the old system
concretion	making informal statements less vague
generalises	property shown for one member of a class and generalised to all others
an-instance-of	properties shown for all components of a certain class

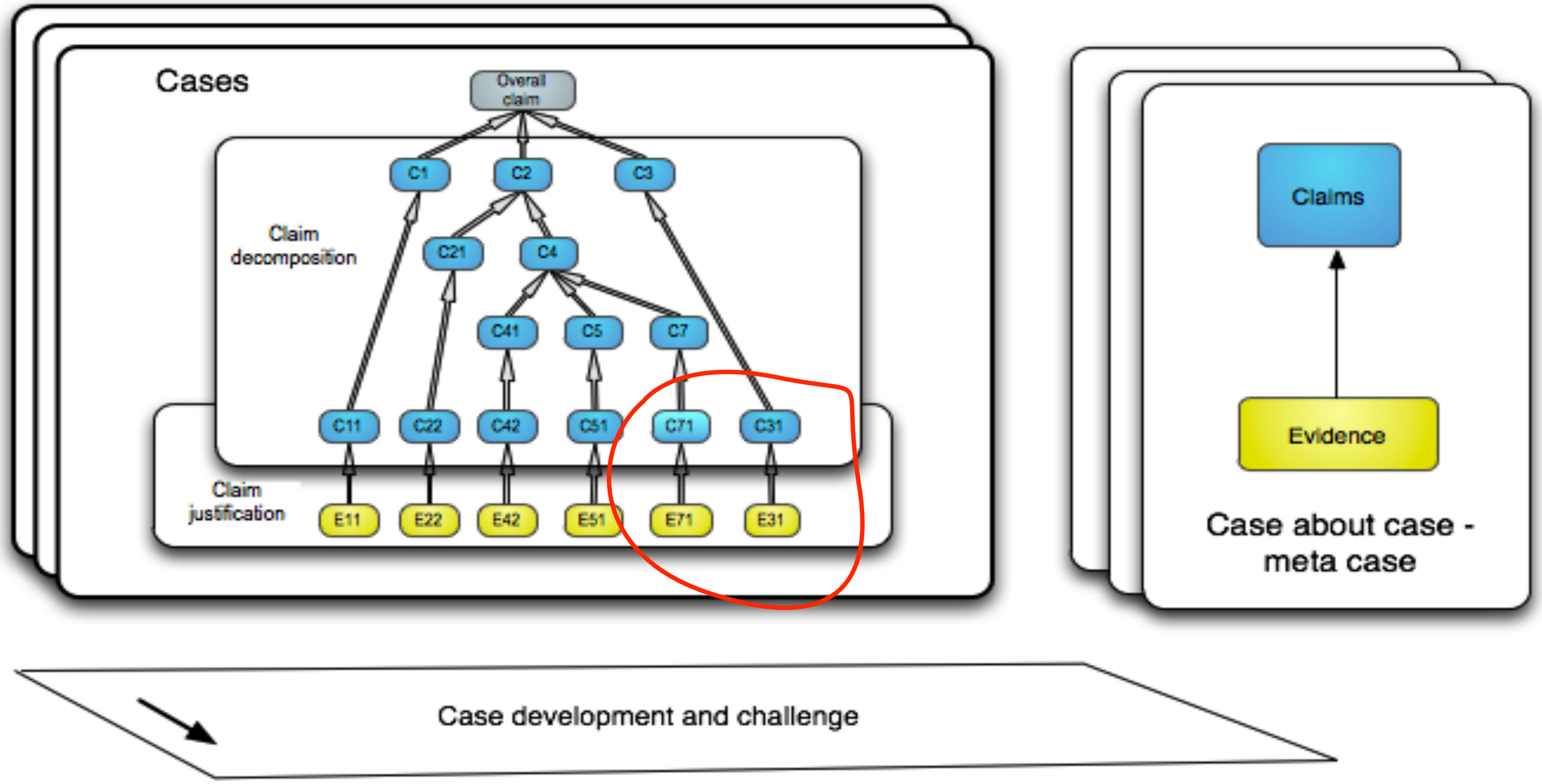
Argument metaphors

- Architecture of cases
- There is a parallel between architecture and argument structure
- e.g. in use of diversity, single failure criterion, sensitivity studies
- metaphors of “belt and braces”, “legs to stand on”
- formalisation difficult and current research topic



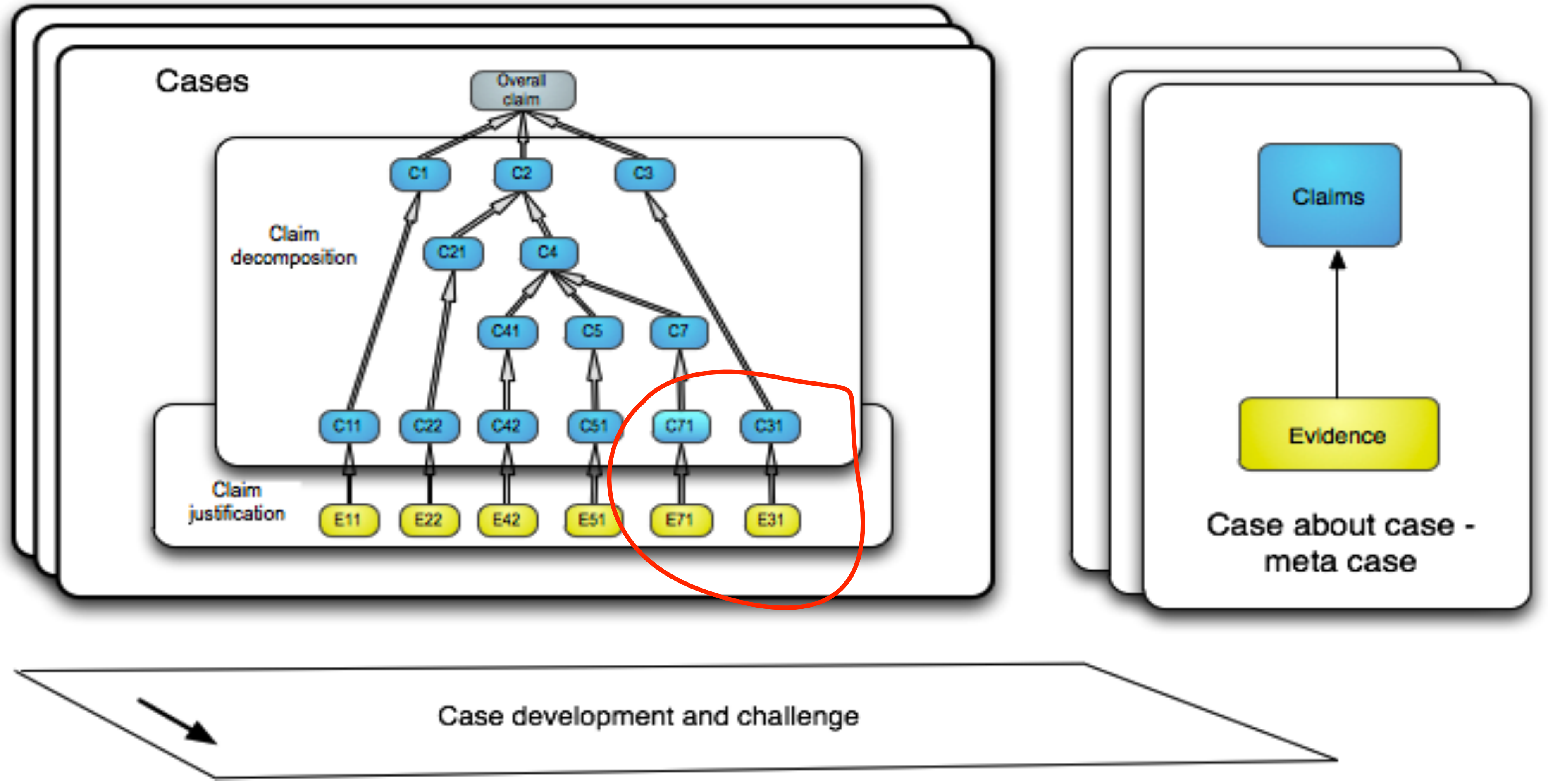
Map evidence to claims

- iterative selection of techniques that generate evidence



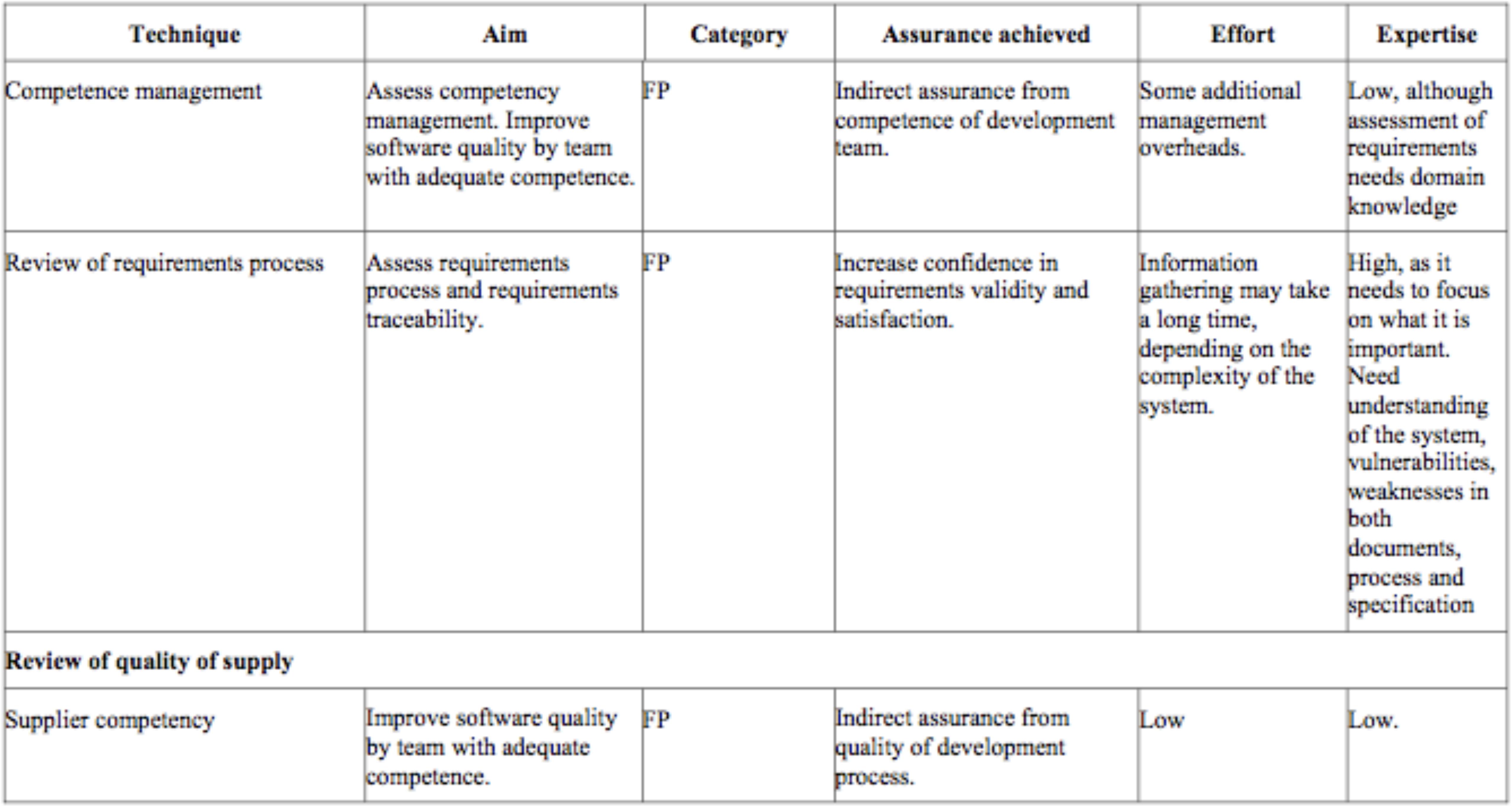
Map evidence to claims

- iterative selection of techniques that generate evidence



Selecting techniques and activities to generate evidence

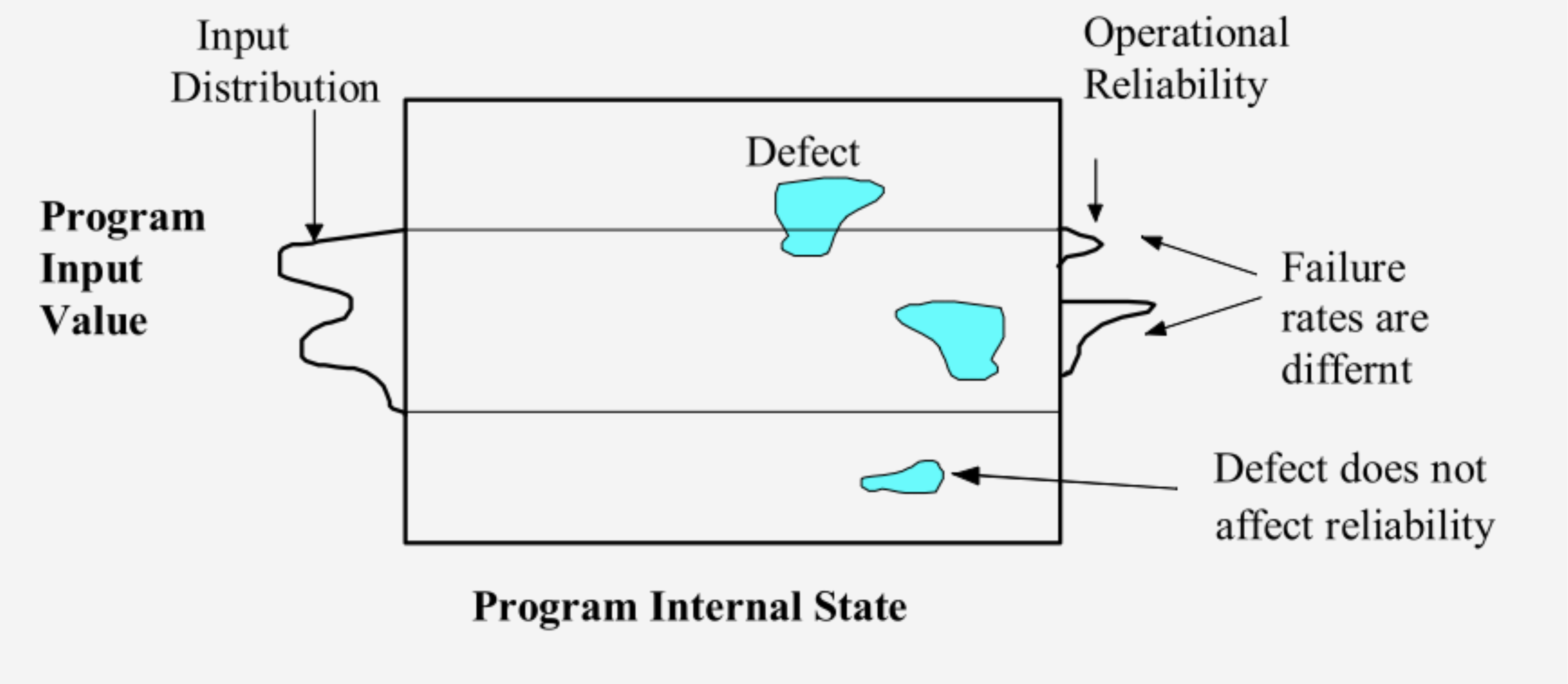
- Catalogues of techniques e.g. in IEC 61508 Part3
  - P Bishop book
- Standards leave it as “exercise for the reader” in justifying selection
  - Supported by case
- Two useful mappings are
  - Activities/techniques → role in case
  - Attributes -> techniques
- Examples tables



Technique	Aim	Category	Assurance achieved	Effort	Expertise
Competence management	Assess competency management. Improve software quality by team with adequate competence.	FP	Indirect assurance from competence of development team.	Some additional management overheads.	Low, although assessment of requirements needs domain knowledge
Review of requirements process	Assess requirements process and requirements traceability.	FP	Increase confidence in requirements validity and satisfaction.	Information gathering may take a long time, depending on the complexity of the system.	High, as it needs to focus on what it is important. Need understanding of the system, vulnerabilities, weaknesses in both documents, process and specification
Review of quality of supply					
Supplier competency	Improve software quality by team with adequate competence.	FP	Indirect assurance from quality of development process.	Low	Low.

Reliability and process models

The software failure process

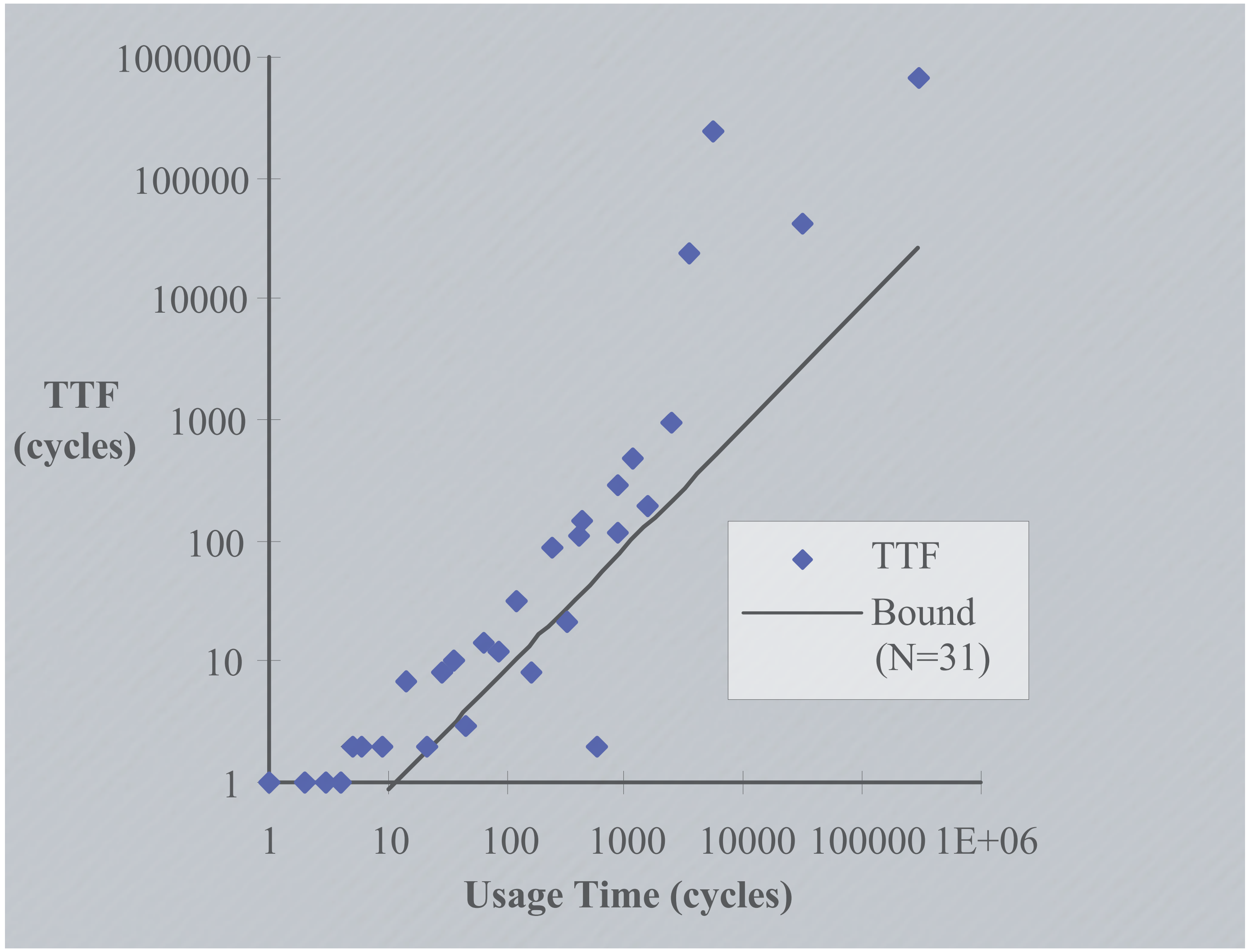


- stochastic nature from sampling input space
- “paradox” of deterministic yet stochastic in behaviour

Conservative long term prediction

$$MTTF_T > e.T / N.d$$

Confirms every engineers intuition



P.G. Bishop and R.E. Bloomfield, ÒA Conservative Theory for Long-Term Reliability Growth PredictionÓ IEEE Trans. Reliability, vol. 45, no. 4, Dec. 96

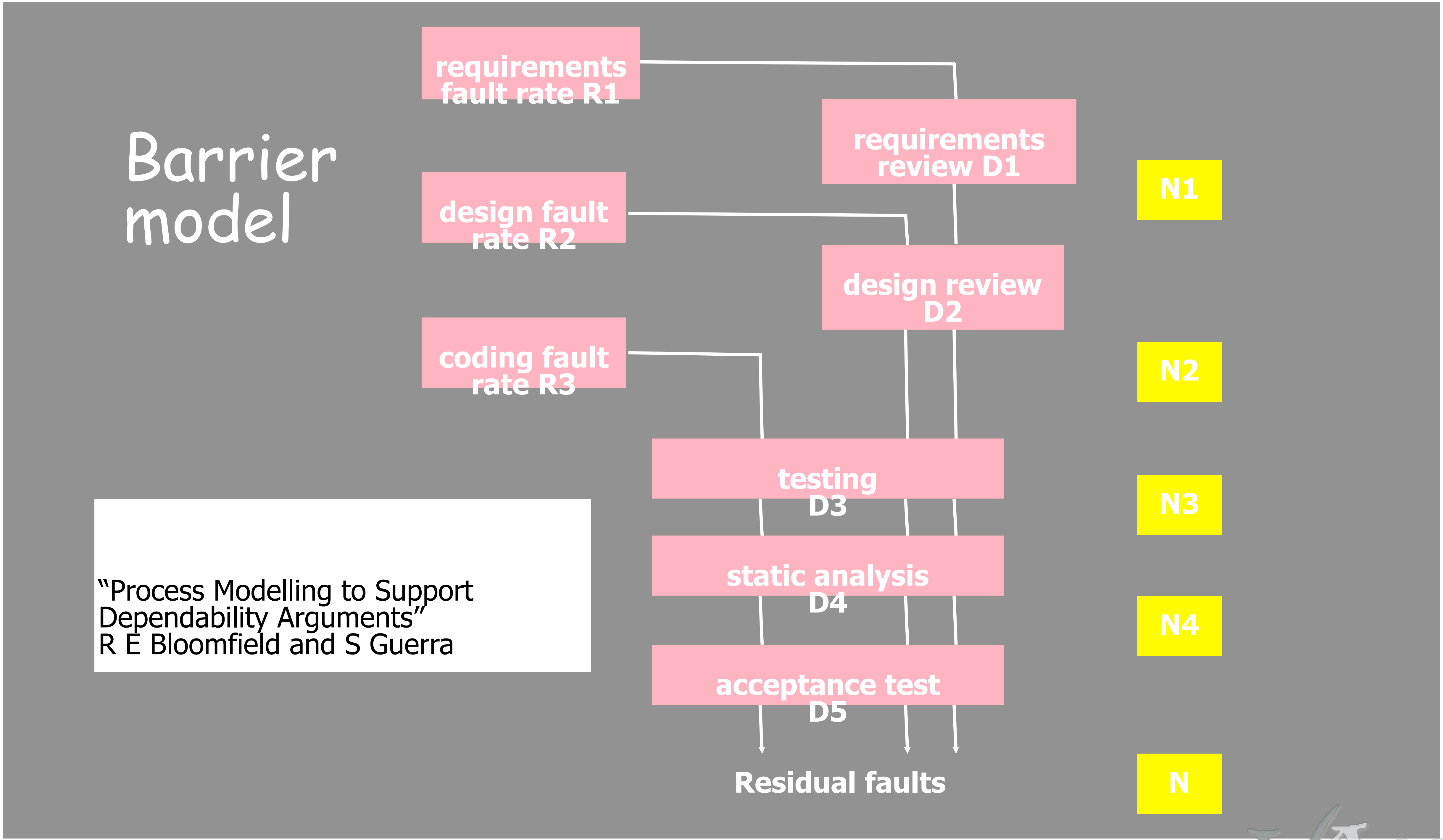
CSR Building confidence in a computerised world

www.csr.city.ac.uk

Adelard

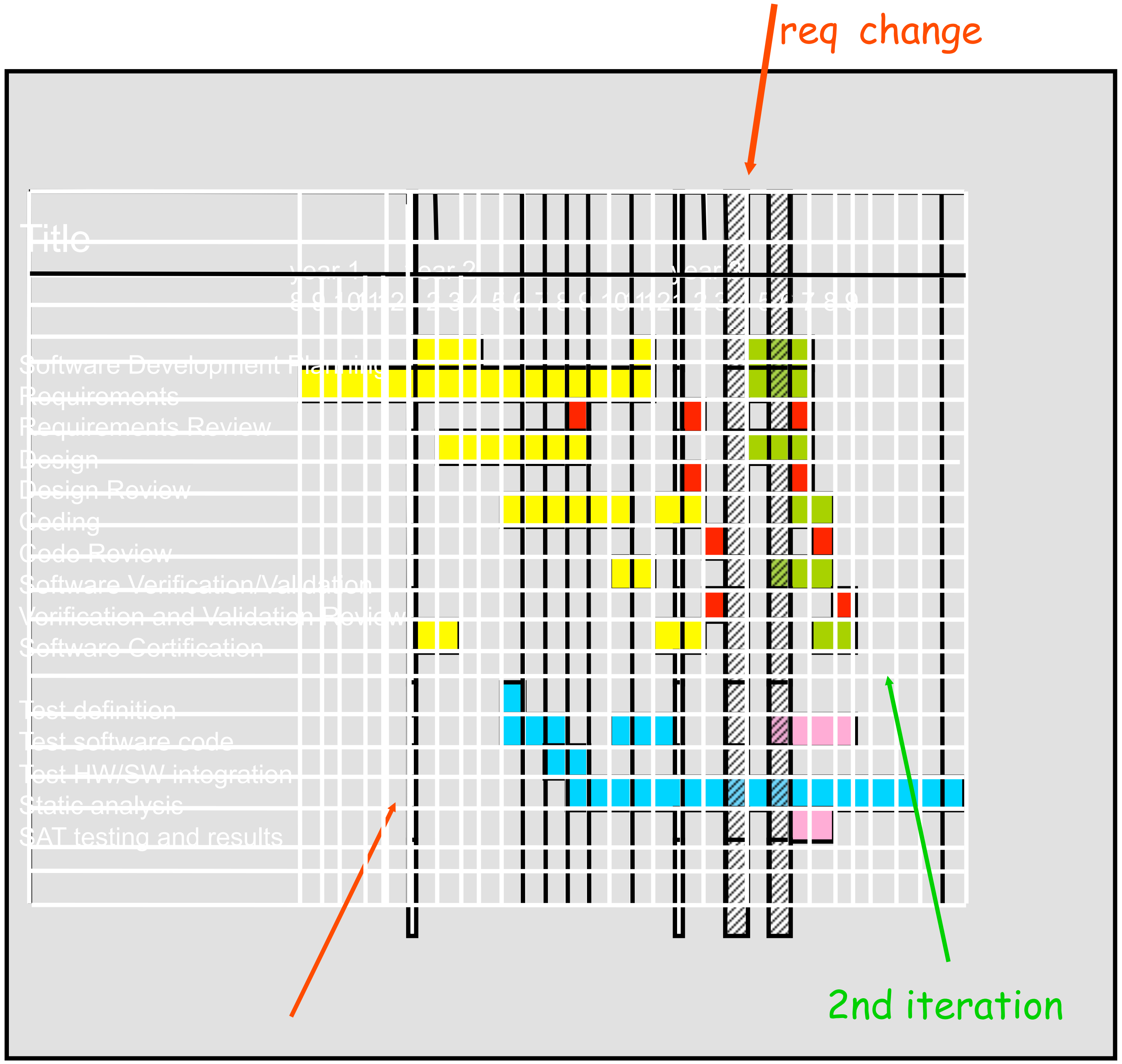
Software development process

Barrier model



"Process Modelling to Support Dependability Arguments"
R E Bloomfield and S Guerra

Use the results of the modelling



req change

2nd iteration

1st iteration

- Estimate residual faults.
- Reliability prediction techniques.
- Identification of weak areas in the process.
- Aiding process improvement
- Explore hypothesis as:
  - “what happens if design fault detection is increased to 90% by the use of tool xyz?”

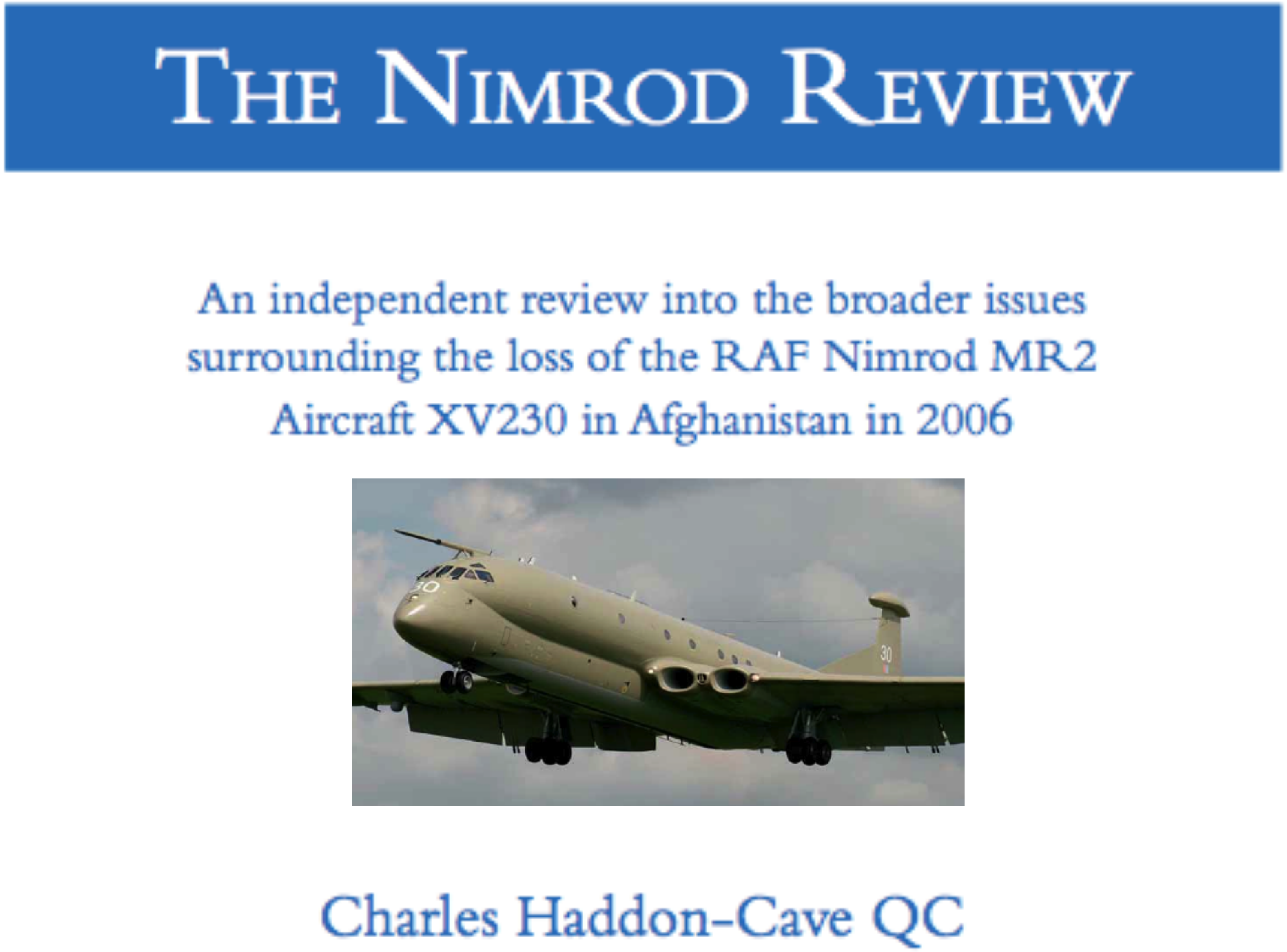
Is this enough?

- If we have a claim decomposition that we think is adequate
- Is this enough?

Evidence

Can we trust the evidence?

Can we trust evidence?

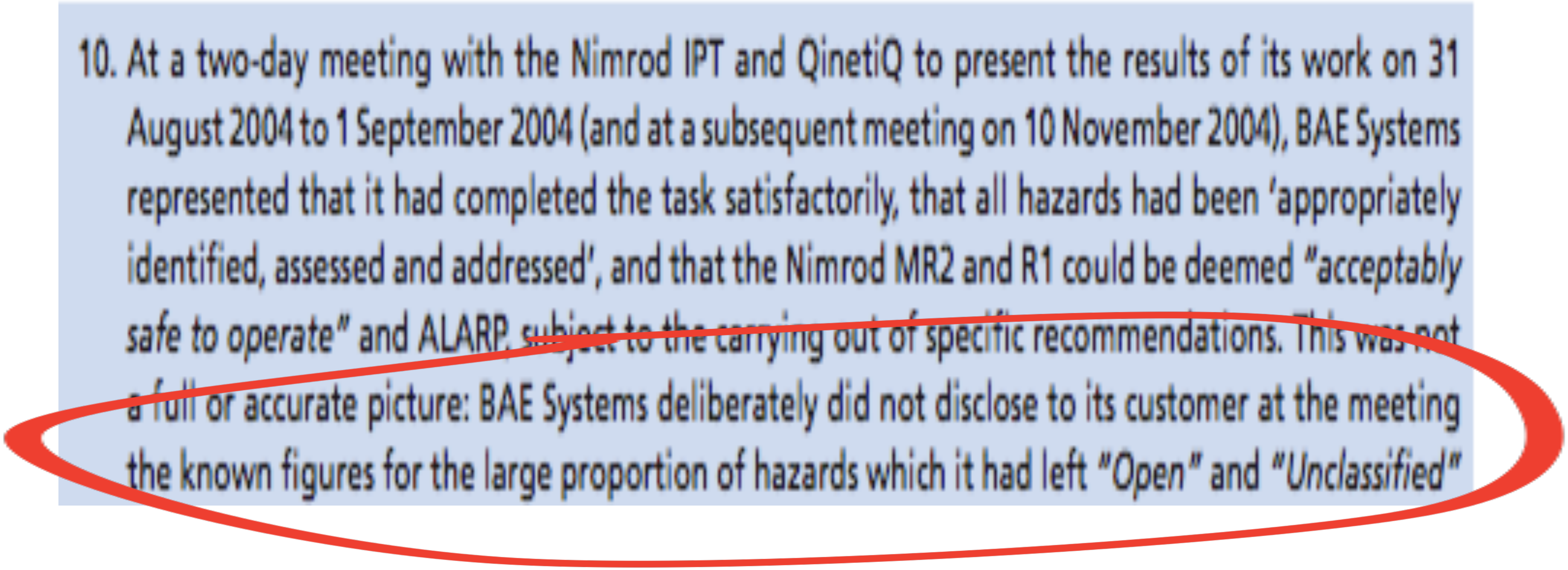


THE NIMROD REVIEW

An independent review into the broader issues surrounding the loss of the RAF Nimrod MR2 Aircraft XV230 in Afghanistan in 2006

Charles Haddon-Cave QC

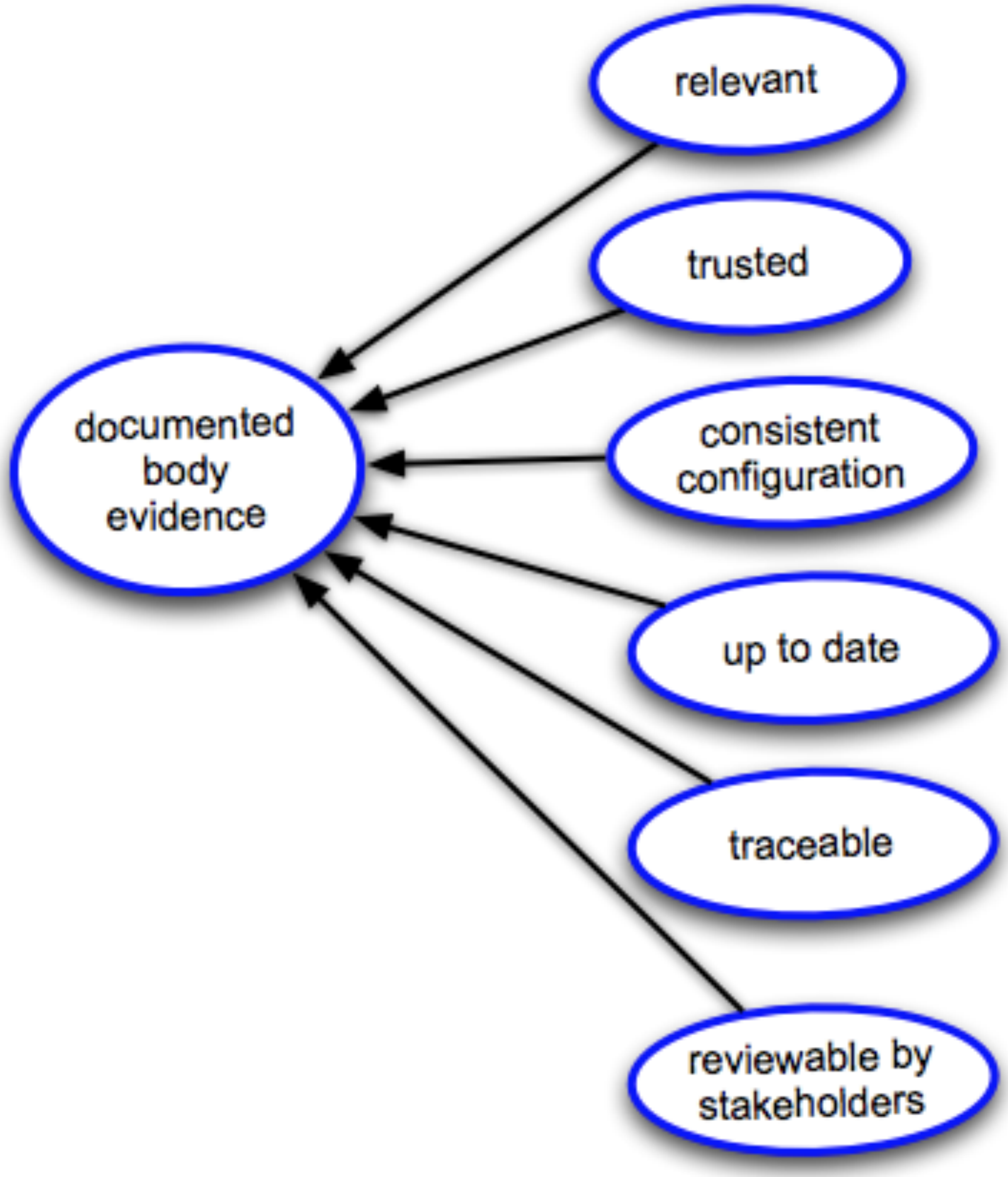
Can we trust the evidence?



10. At a two-day meeting with the Nimrod IPT and QinetiQ to present the results of its work on 31 August 2004 to 1 September 2004 (and at a subsequent meeting on 10 November 2004), BAE Systems represented that it had completed the task satisfactorily, that all hazards had been 'appropriately identified, assessed and addressed', and that the Nimrod MR2 and R1 could be deemed *"acceptably safe to operate"* and ALARP, subject to the carrying out of specific recommendations. This was not a full or accurate picture: BAE Systems deliberately did not disclose to its customer at the meeting the known figures for the large proportion of hazards which it had left *"Open"* and *"Unclassified"*

Evidence

"a documented body of evidence that provides a convincing and valid argument that a system is adequately safe for a given application in a given environment"



Evidence

- Know it applies to the system we are evaluating
  - Configuration consistency
  - Not trivial, related to threat assumptions
- Trust organisation that is providing it
- Traceable to process, tools and people that produced it
- Relevant, not information on a truck
- Sufficiently detailed
- Continues to be trusted
  - Changes to tools, systems
  - Knowledge management
- Accessible

Meta-case

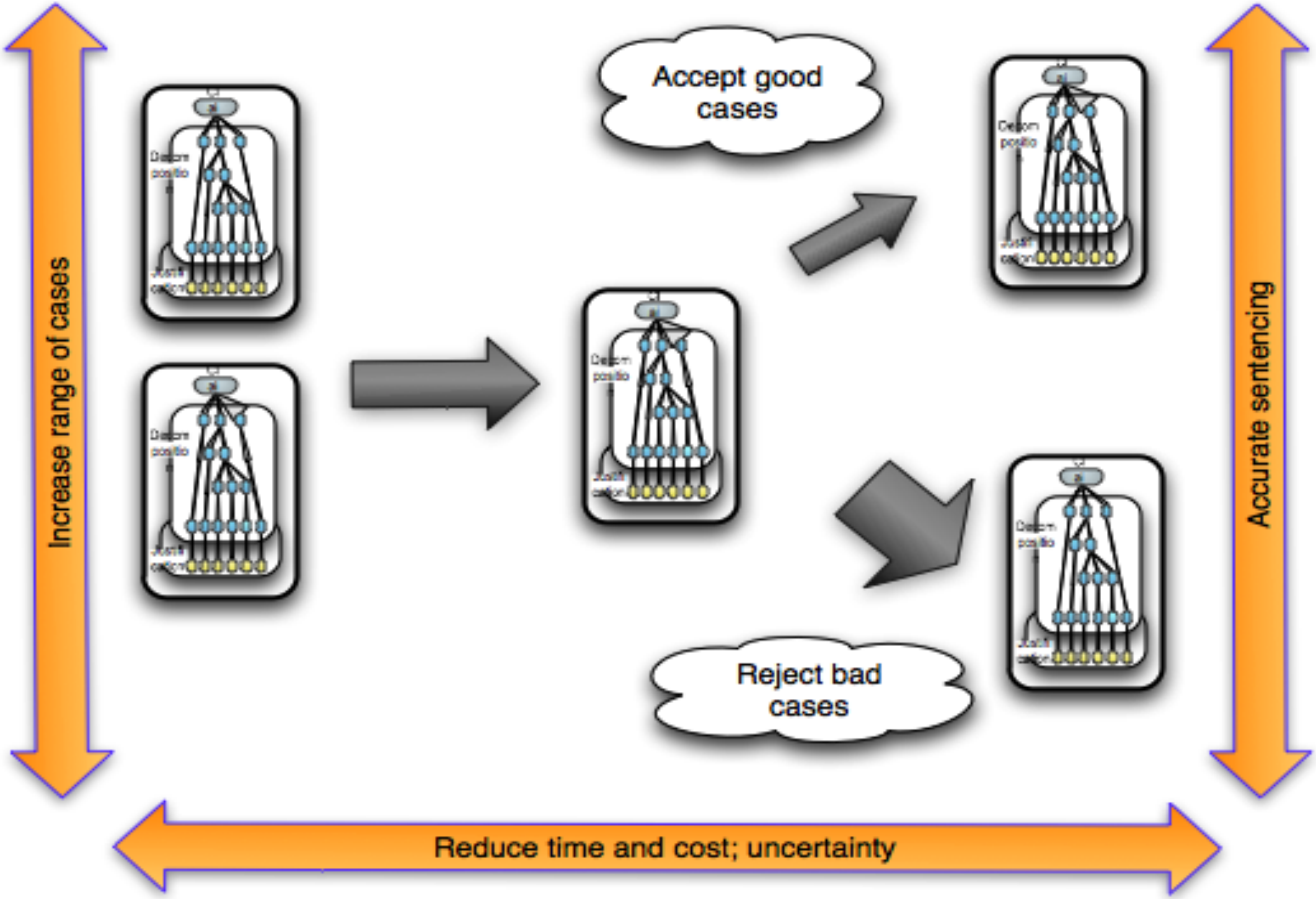
Research and development landscape

Research and development

- Structures and scope of cases
  - How to justify the structure
  - Use of formal structures
  - Structures for different types of COTS components
  - Compositionality
  - Socio-technical perspective
  - Security, resilience and other cases
- Risk communication and scalability
- Role of standards
  - How to integrate standard compliance arguments
- Model based System/hazard analysis

- Styles of cases
  - Black-box
  - LowSIL
- Systems and cases
  - Architectures
  - Diversity
- Stopping rules
  - Claim limits and justification of numerical claims
- Confidence
- Evidence generation
  - Techniques and software analysis
  - Focused proof
  - Combing static/dynamic

Objectives of research



Increase range of cases

Accept good cases

Reject bad cases

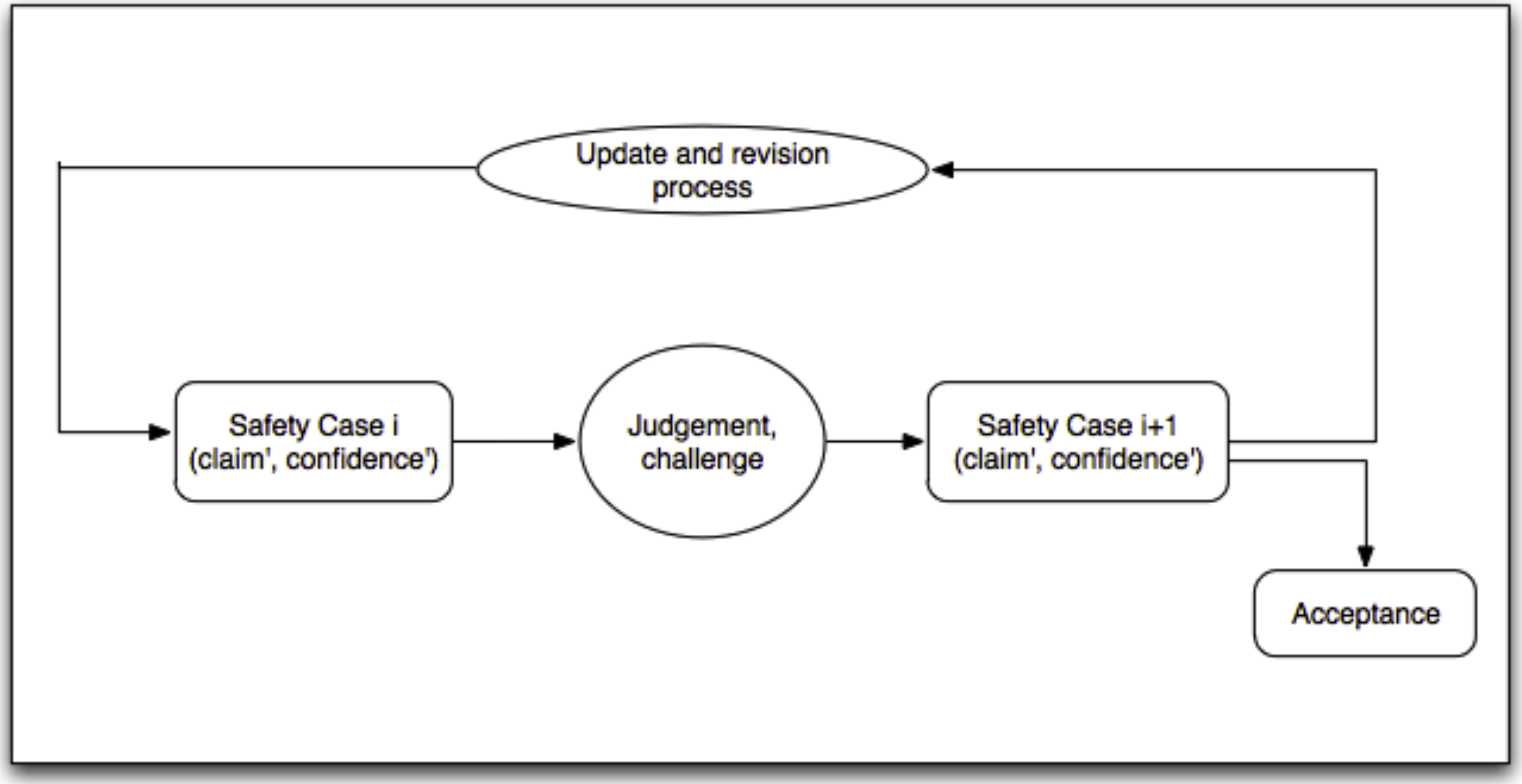
Accurate sentencing

Reduce time and cost; uncertainty

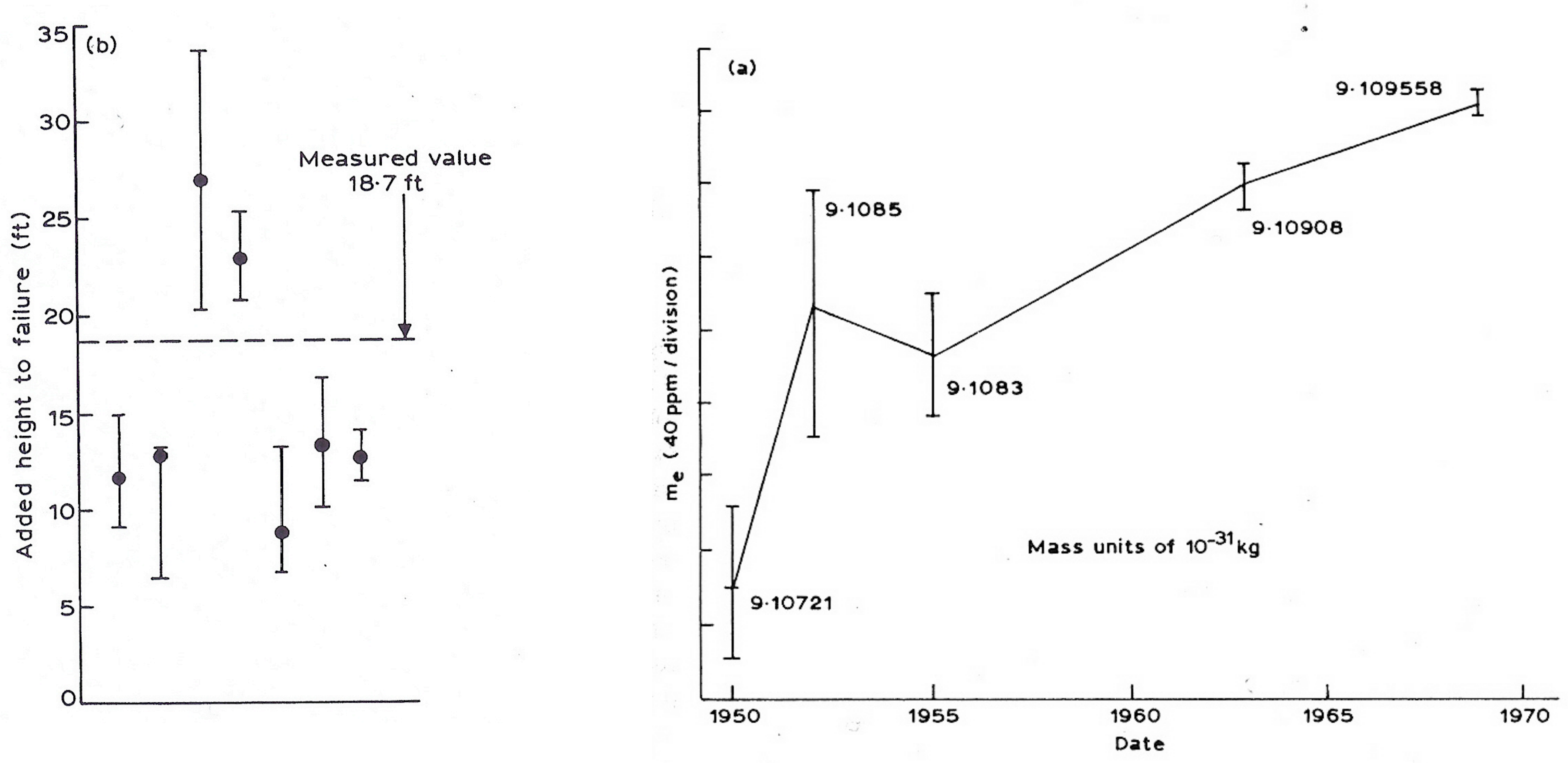
Confidence

Safety case process – building confidence

- Captured in safety management system and in meta-case
- Challenge and response cycle essential


Update and revision process
Safety Case i (claim', confidence')
Judgement, challenge
Safety Case i+1 (claim', confidence')
Acceptance

Confidence in physics and engineering



from Henrion and Fishcoff, also see “How experiments end”, Peter Galison,Chicago 1987

There are two sources of uncertainty…

- There is uncertainty about when a system will fail
  - In the jargon: ‘aleatory uncertainty’
  - It is now widely accepted that this uncertainty should be expressed probabilistically (e.g. failure rate, pfd, etc)
- There is uncertainty about the reasoning used to support a dependability claim
  - In the jargon: ‘epistemic uncertainty’
  - In particular, the role of expert judgment
  - The appropriate calculus here is Bayesian (subjective) probability

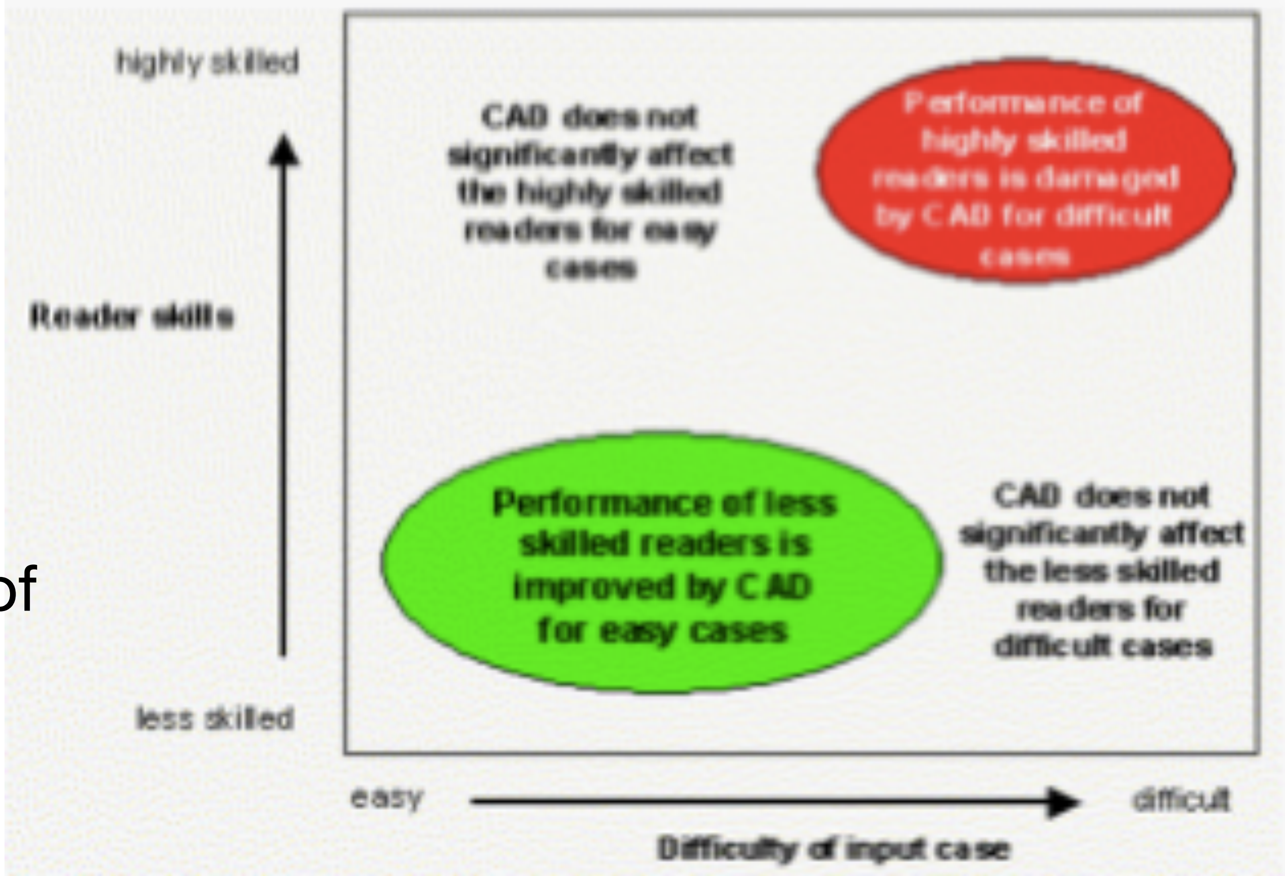
Work on confidence - summary

- Interpret existing practice in terms of confidence
  - Nuclear SAPS, ACARP in SOUP and SOCS report, CAA Regulatory oversight
- Empirical short study on assessors and SIL judgements
- Modelling of confidence in SILS, show impact, concepts and make speculative advice on standards.
- Confidence and legs (Littlewood, Bloomfield DSN)
- Extensive analysis of simple BBNs (Littlewood and Wright)
- Theoretical work on conservative approach, and later more useful bounds (TSE)
- Aleatory and epistemic distinction and dealing with system architecture/argument structures (Littlewood and Rushby)
- Threat models
- Stress claim/confidence pairs

Socio-technical

- A socio-technical perspective on assurance cases:
  - In addition to claims that physical hazards, security threats have been addressed
  - Define a range of vulnerabilities (narrow scope, misaligned responsibilities, undifferentiated users, adaptation, automation biases, non-independence of arguments) and develop arguments of how they might be addressed.
  - Develop methods for review wrt socio-technical issues

Ideas taken from EPSRC INDEED and DIRC projects


highly skilled
Reader skills
less skilled
CAD does not significantly affect the highly skilled readers for easy cases
Performance of highly skilled readers is damaged by CAD for difficult cases
Performance of less skilled readers is improved by CAD for easy cases
CAD does not significantly affect the less skilled readers for difficult cases
easy
difficult
Difficulty of input case

Supply chain examples

Supply chain examples

An approach to cases for the nuclear industry

UK nuclear industry

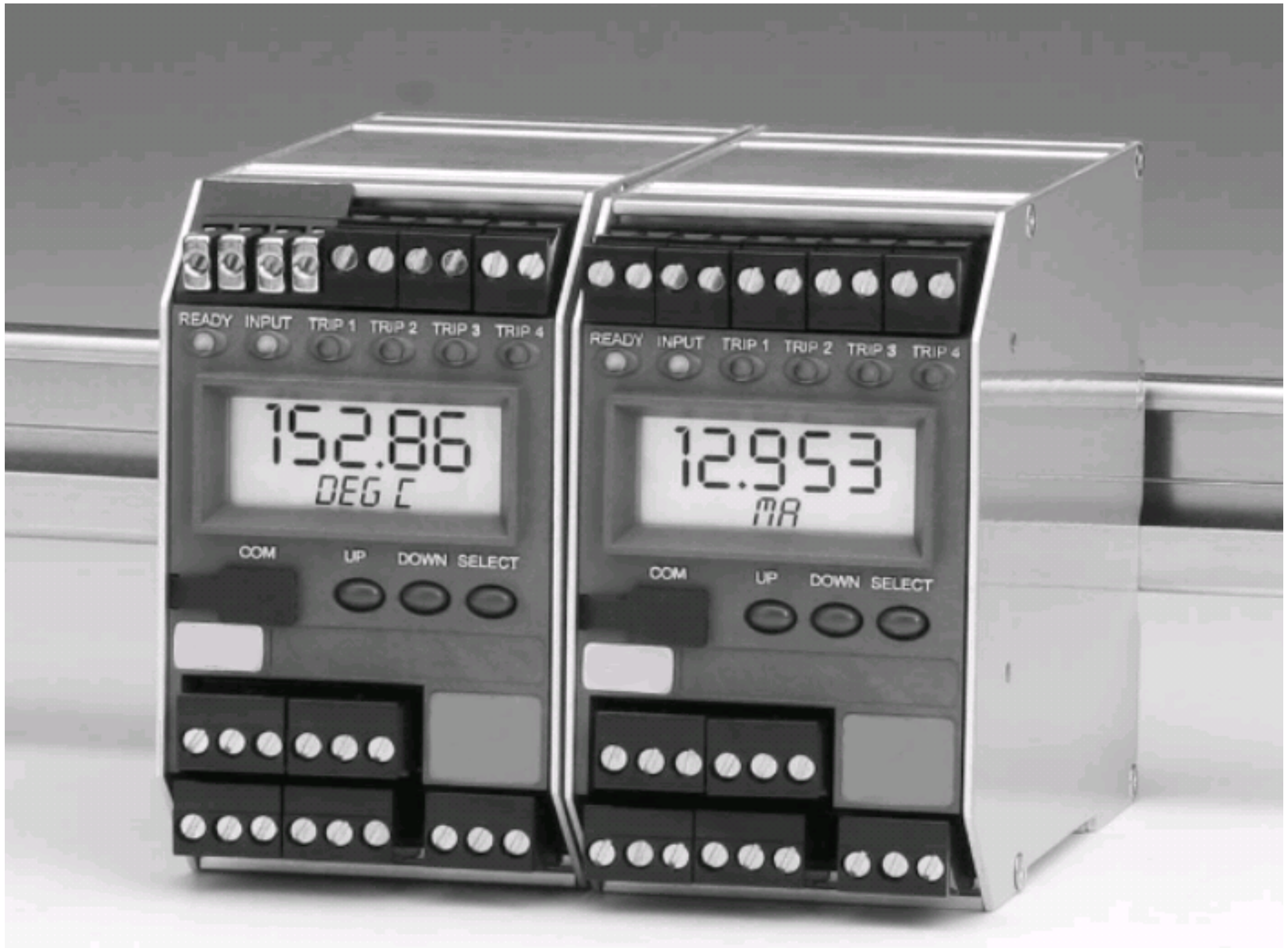
- Drivers
  - Intense interest in “New build”. Regulatory requirements expressed in terms of claims, arguments, evidence
  - Ageing nuclear plant being life extended, older simple technology being replace by smart sensor and actuators. Relative small user, but advantages and necessity of using smart devices
  - Two parts to strategy
    - dialogue with supply side, building trust or at least understanding
    - technical approaches to assessment, add value to supplier and user
- Context of need to show
  - compliance with standards, reality of non-compliances
  - principled approach to addressing these, wrappers, argument strategies, analysis

Safety cases – regulatory obligation

- Safety cases are required by licence conditions
- The Conditions are non-prescriptive and set goals that the licensee is responsible for meeting, amongst other things by applying detailed safety standards and establishing safe procedures for the facilities.
- A "safety case" is defined as
    - the document or documents produced by the licensee documentation to justify safety during the design, construction, manufacture, commissioning, operation and decommissioning phases of the installation.
- Safety Assessment Principles (SAPs) describe safety case process and principles to be covered
    - “.... the establishment of and compliance with appropriate standards and practices throughout the software development life-cycle should be made, commensurate with the level of reliability required, by a demonstration of ‘production excellence’ and ‘confidence-building’ measures.”

Smart sensors

- Special purpose embedded computer systems: replacements for analogue level alarms and transmitters, with more intelligence, connectivity.



The challenges of COTS software

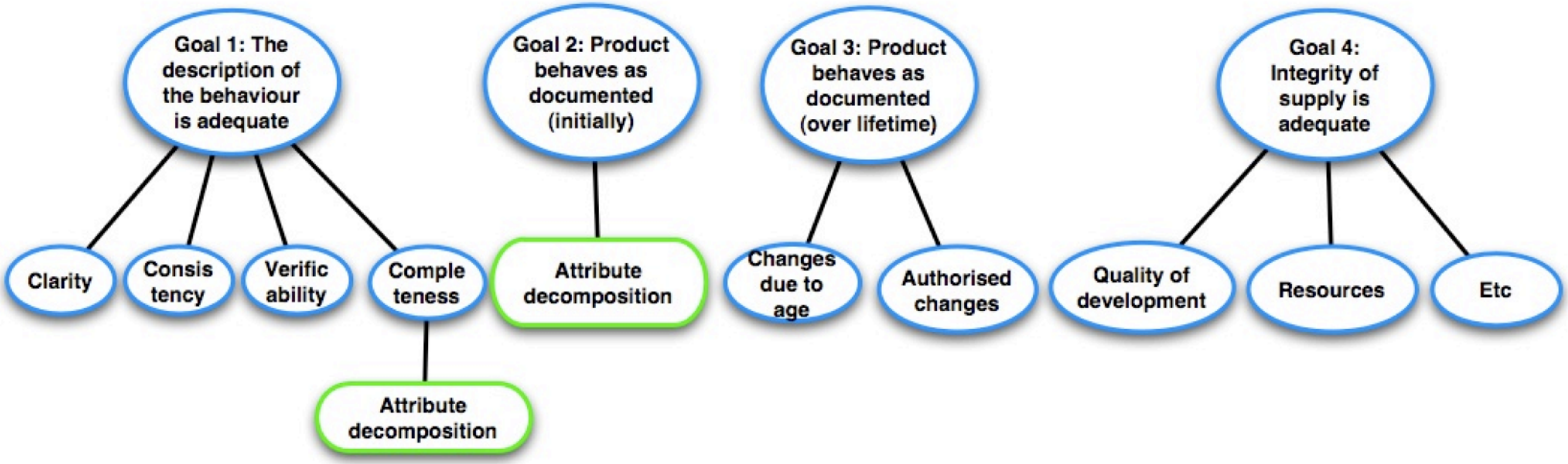
- It has already been written
  - (it’s too late to do it better)
  - We need to be able to work with what we’ve got
- It followed an ordinary industrial process
  - (they didn’t use the B method)
  - We need to be able to reconstruct formal specifications and understand them
- Perfectly reasonable trade-offs were made
  - (coding style for space, for example)
  - We need to be able to accommodate less-than-ideal code

Goal based framework

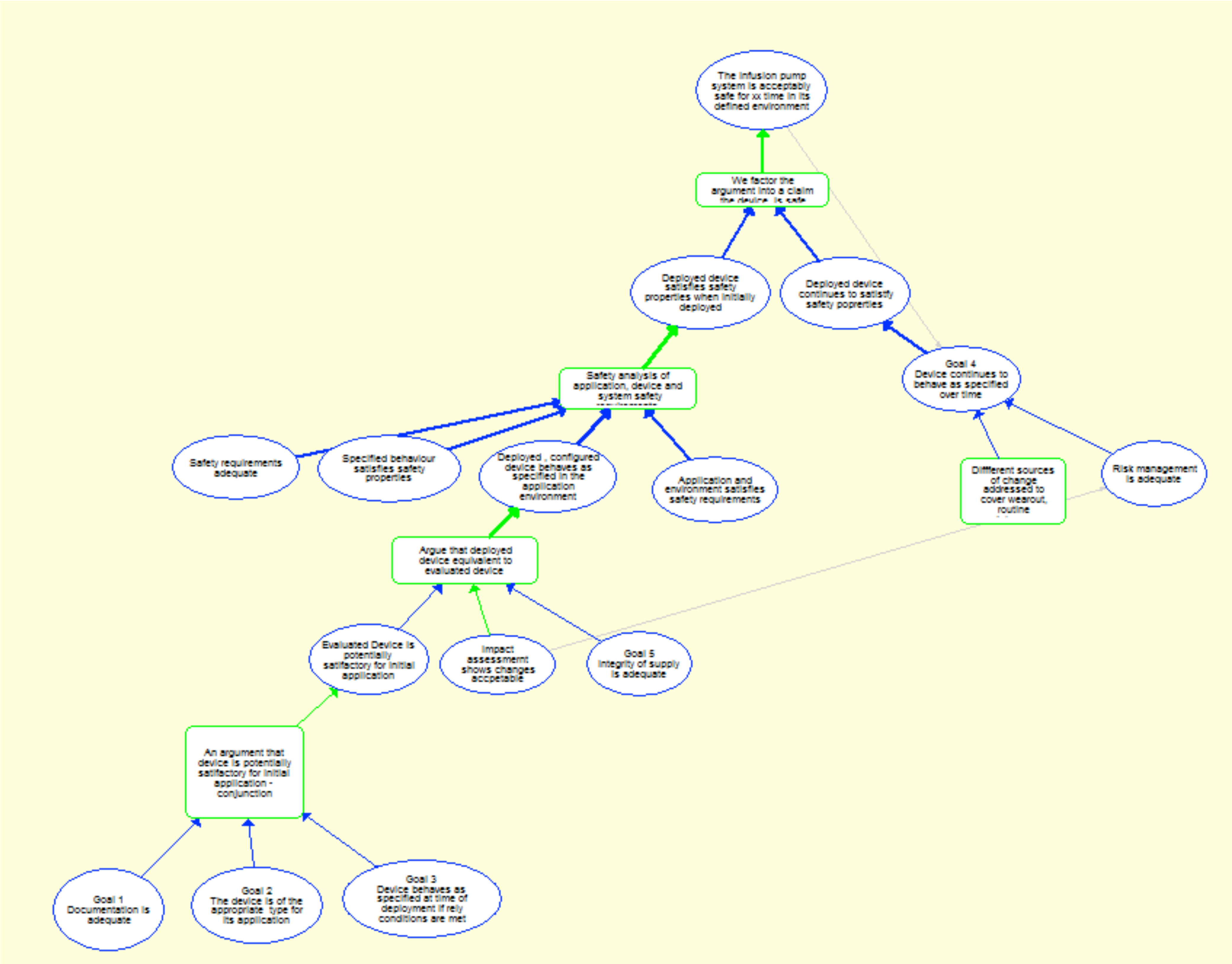
Goal 1: The description of the behaviour is adequate
Goal 2: Product behaves as documented (initially)
Goal 3: Product behaves as documented (over lifetime)
Goal 4: Integrity of supply is adequate

- Goal 1 – the description of the behaviour is adequate
- Goal 2 – product behaves as documented (initially)
- Goal 3 – continues to behave as documented over its lifetime
- Goal 4 – integrity of supply is adequate

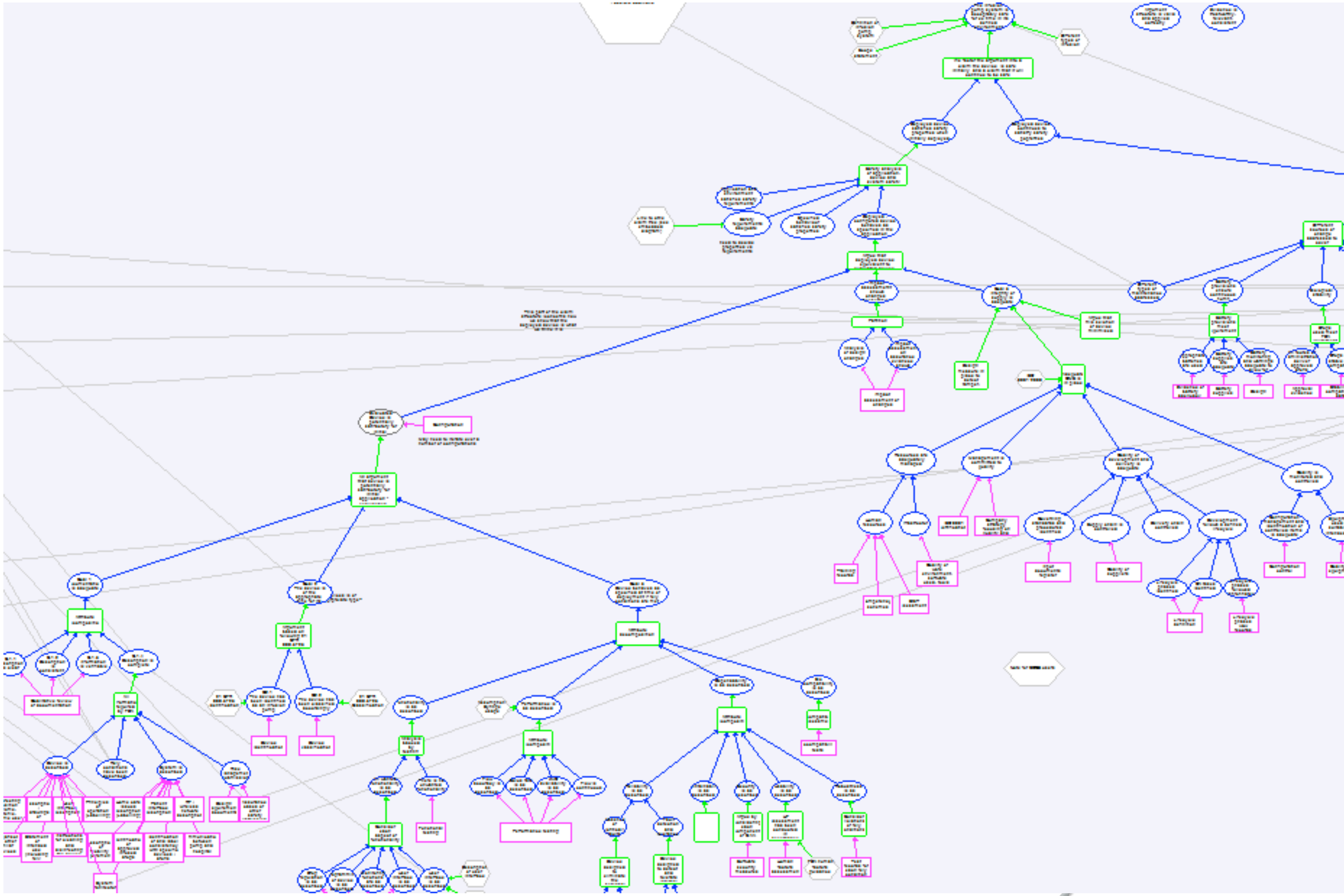
Progressively expanded


Goal 1: The description of the behaviour is adequate
Goal 2: Product behaves as documented (initially)
Goal 3: Product behaves as documented (over lifetime)
Goal 4: Integrity of supply is adequate
Clarity
Consis tency
Verific ability
Comple teness
Attribute decomposition
Changes due to age
Authorised changes
Quality of development
Resources
Etc
Attribute decomposition

Example


The infusion pump system is acceptably safe for xx time in its defined environment
We factor the argument into a claim the device is safe
Deployed device satisfies safety properties when initially deployed
Deployed device continues to satisfy safety poprerties
Safety analysis of application, device and system safety
Goal 4 Device continues to behave as specified over time
Safety requirements adequate
Specified behaviour satisfies safety properties
Deployed , configured device behaves as specified in the application environment
Application and environment satisfies safety requirements
Different sources of change addressed to cover wearout, routine
Risk management is adequate
Argue that deployed device equivalent to evaluated device
Evaluated Device is potentially satifactory for initial application
Impact assessmemt shows changes accpetable
Goal 5 Integrity of supply is adequate
An argument that device is potentially satifactory for initial application - conjunction
Goal 1 Documentation is adequate
Goal 2 The device is of the appropriate type for its application
Goal 3 Device behaves as specified at time of deployment if rely conditions are met

Example

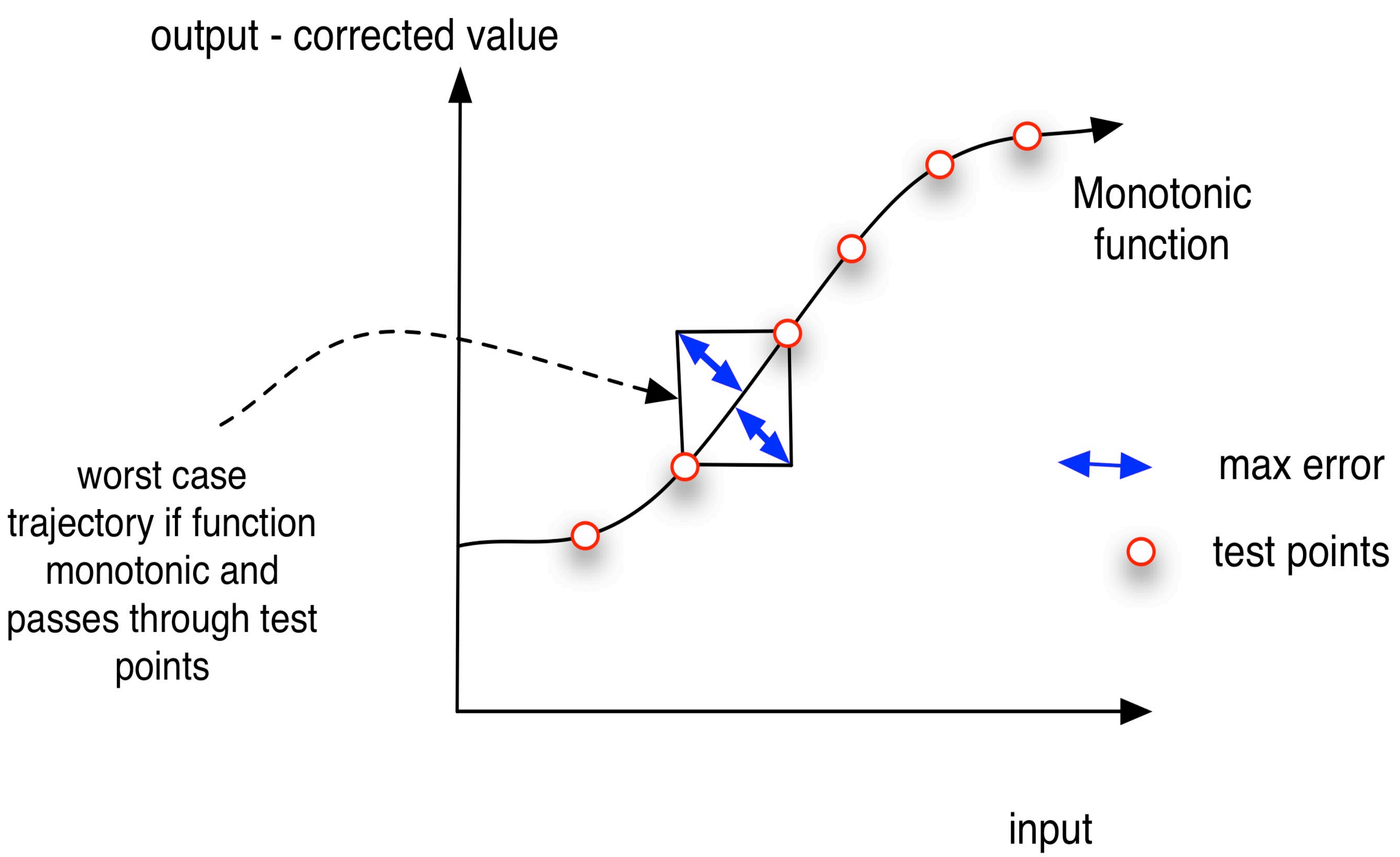


Nuclear example – component justification

- Four goals to justify that component behaves as its description
- Combines three types of arguments
  - Behaviours goals are met
  - Vulnerabilities are identified and mitigated
  - Adequate development process, compliance
- Temporal view
  - Considers lifetime of the product
  - OK when deployed + continues to be OK
- Use of traffic lights to indicate justification of different claims
- Innovative approach for nuclear industry in the UK

Evidence generation and types of arguments

- Analysis of software
  - C and assembler
  - Integrity static analysis
  - Concurrency analysis
  - Failure integrity analysis
  - Focused proof
  - Combining static/dynamic



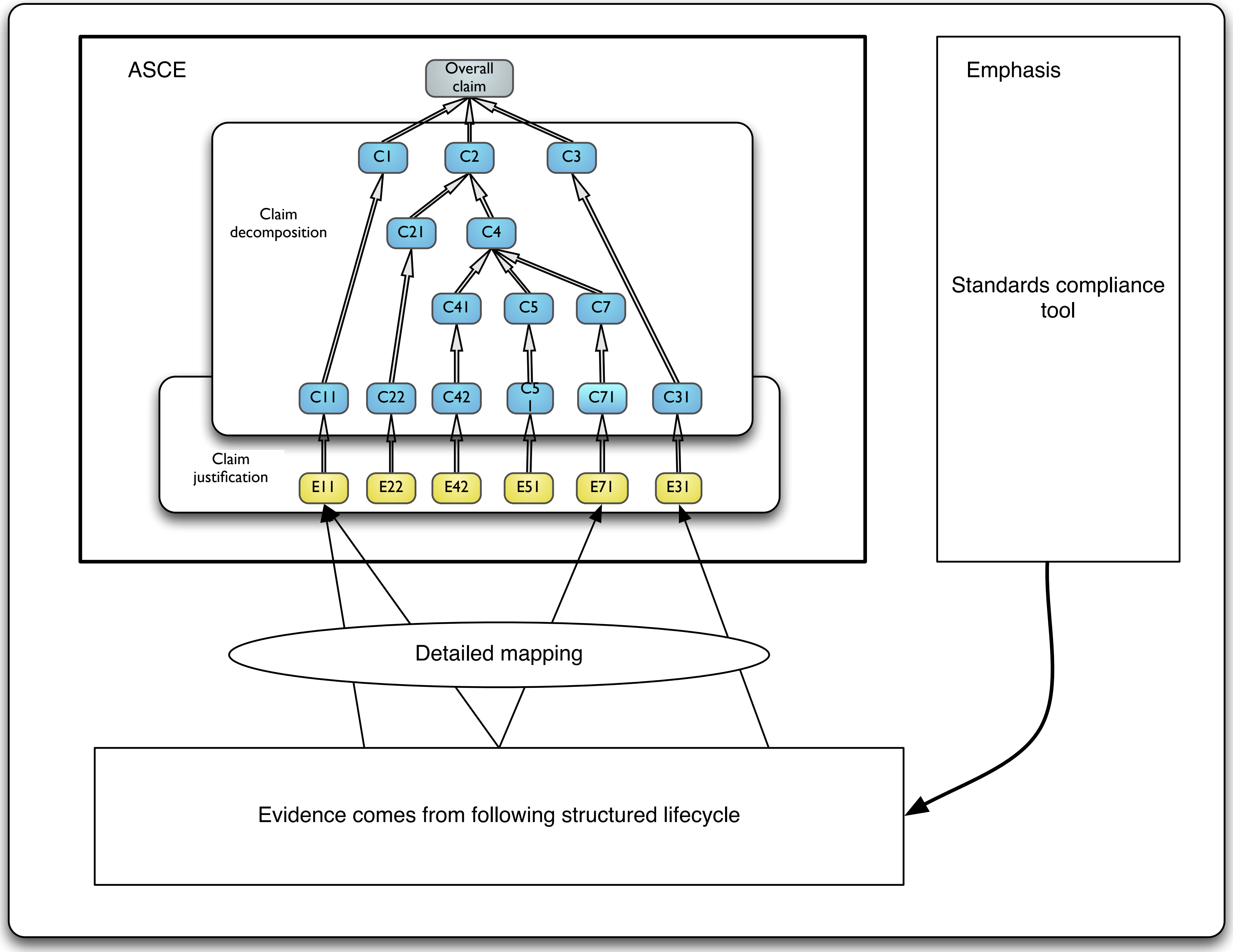
worst case trajectory if function monotonic and passes through test points

Evidence generation

- At Adelard, we've developed a strategy
  - Use modern tools, the Safer C Toolkit, CodeSurfer, Frama-C
  - Extend them where necessary, or build our own tools
  - Use combinations of tools to reduce costs: only use the strongest techniques on the most important code
- We've **driven down the cost** of techniques for smart devices
  - For example, we believe that **formal verification** is now applicable at SIL 2 (not just SIL 4)
  - We can focus our efforts on code that matters
  - We can apply techniques quicker and more effectively
- Approaches to Software Criticality Analysis, integrity analysis, black box, combined static/dynamic analysis

Emerging integrated approach

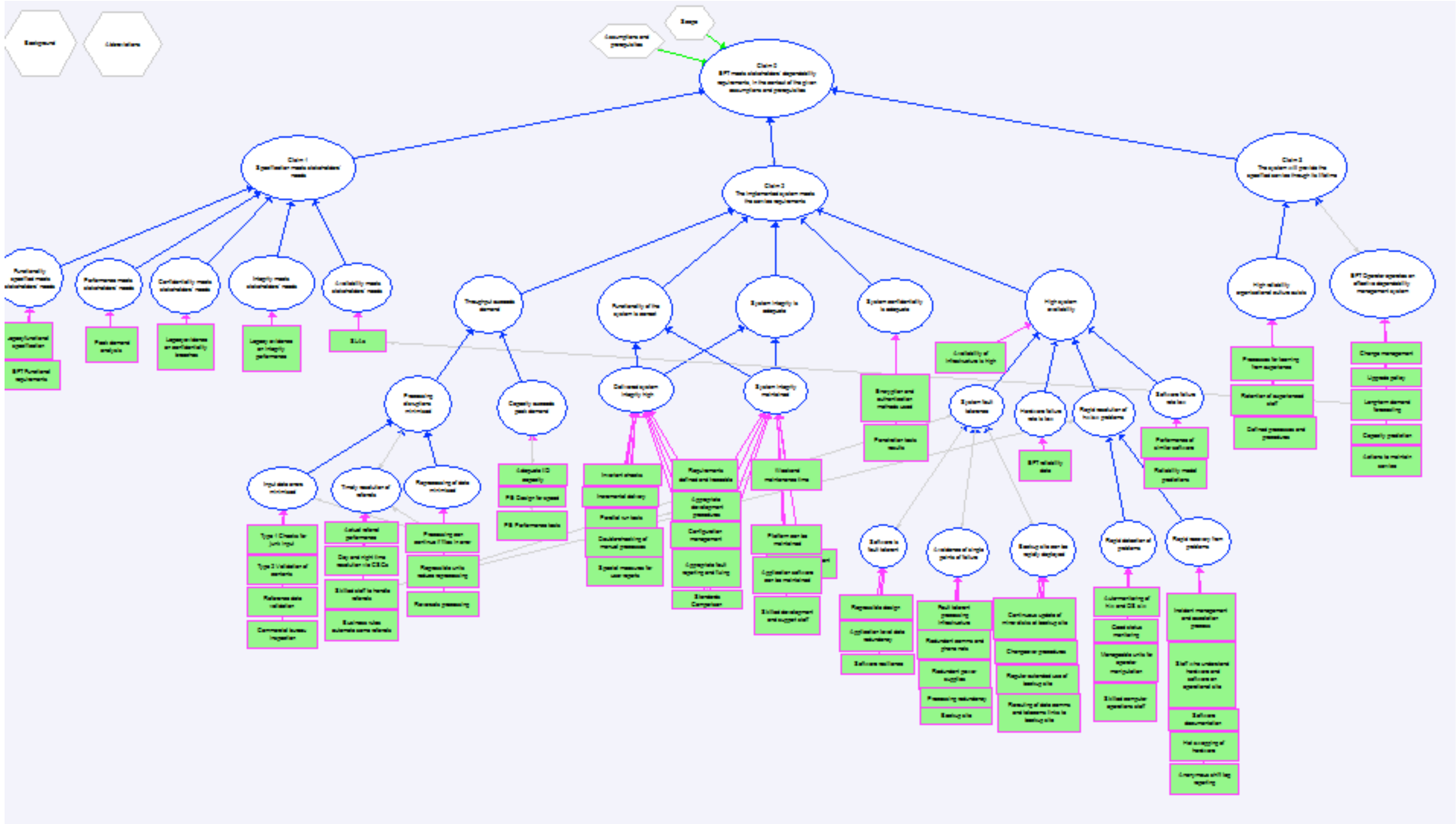
- identify gaps and compensations
- principled approach to non-compliance
- tool supported
- nuclear industry (initially)



Financial services dependability

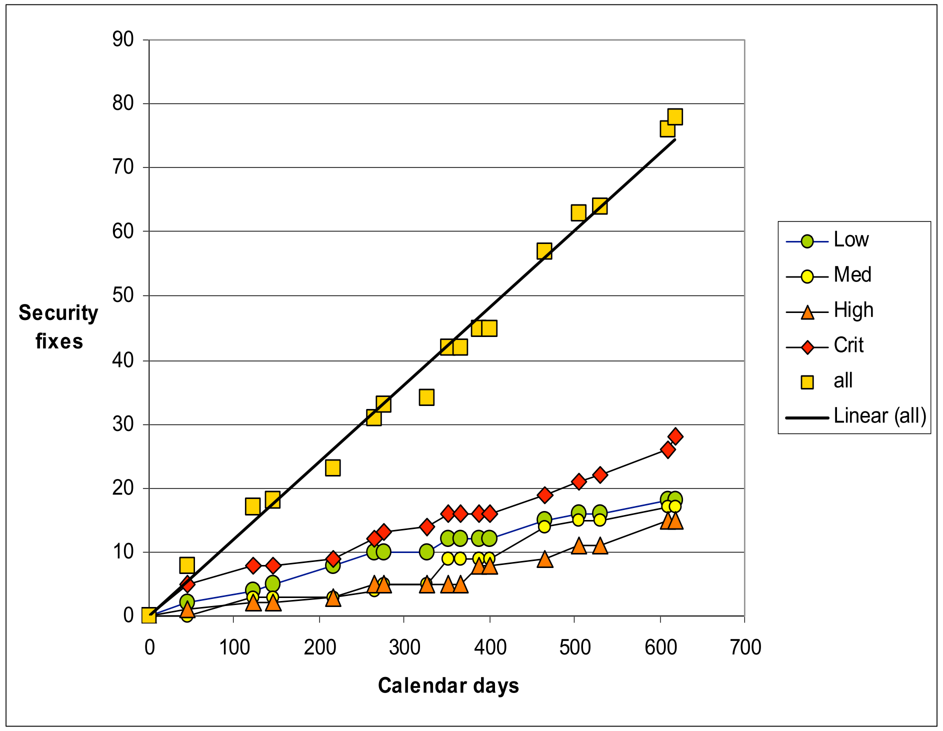
High volume

- Socio-technical perspective
- Deployment decision
- Range of stakeholders



Security engineering

- programme of work on crossover security engineering from safety
- structured assurance cases - service oriented
  - hazard based
  - vulnerability growth models, etc



Service Interface	Confidentiality	Integrity	Availability
Intelligent attack of service interface			
Service user malpractice			
Service user equipment/ software vulnerabilities			

CSR Building confidence in a computerised world

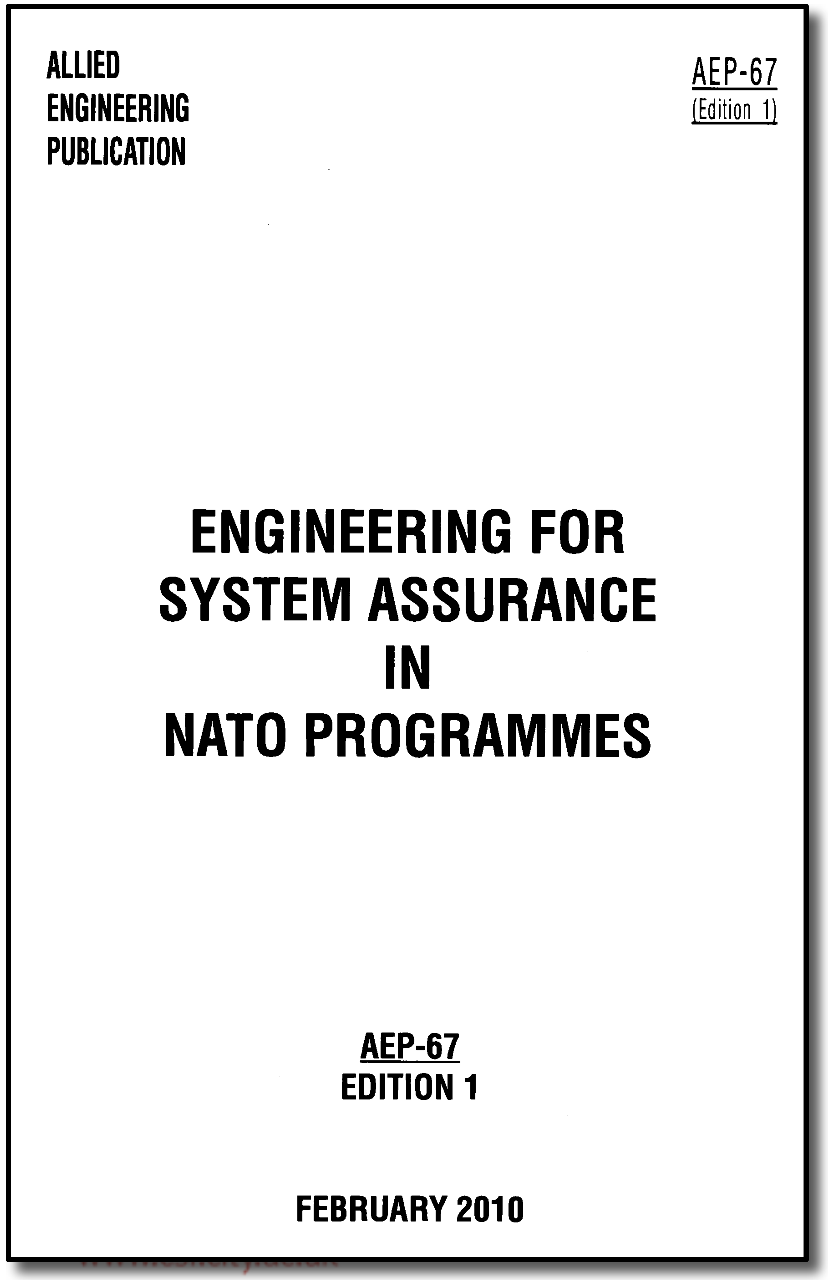
www.csr.city.ac.uk

Adelard

Role of assurance cases in supply chain

Supply chains

- evaluation
- communication



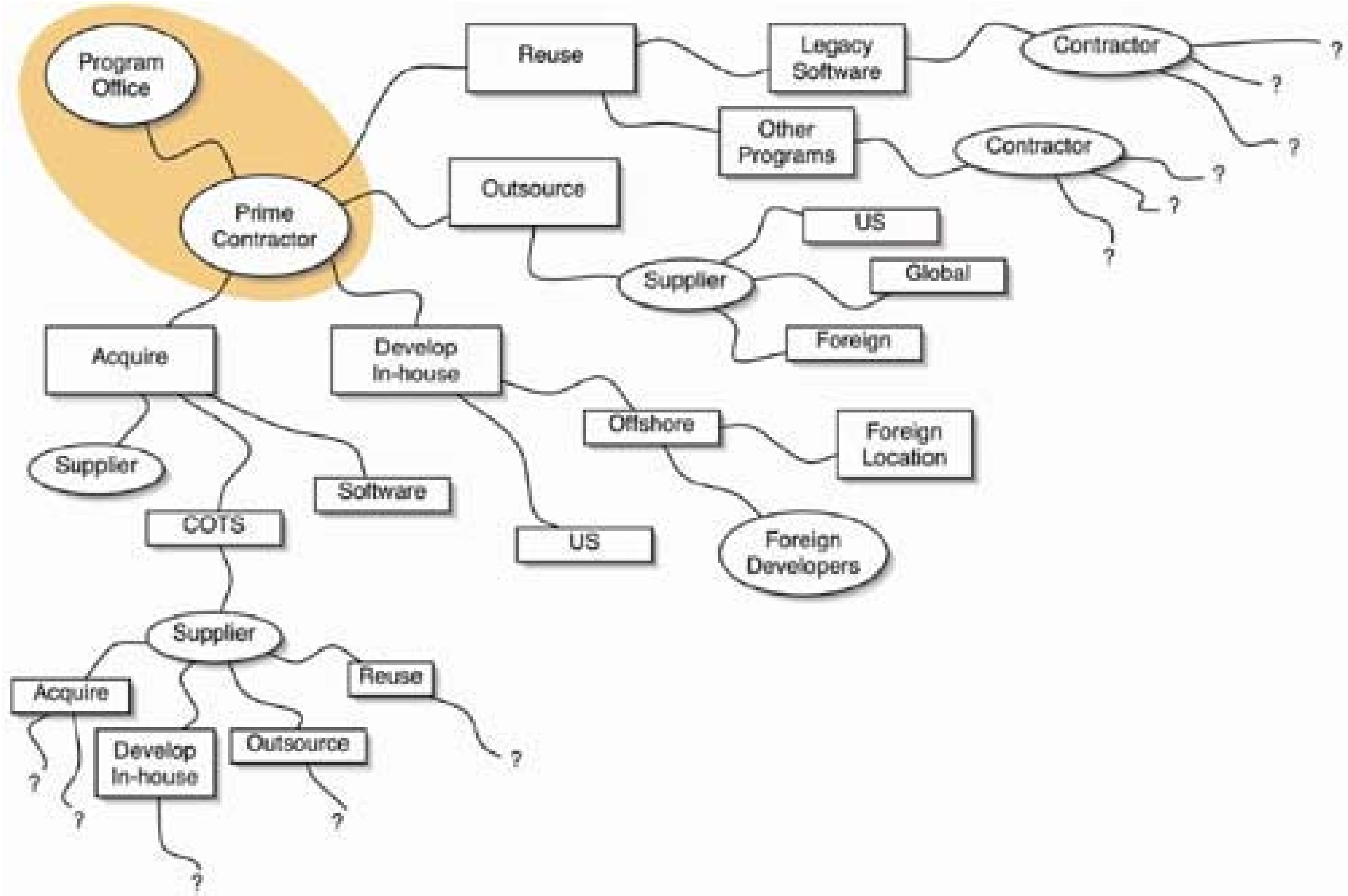
ALLIED
ENGINEERING
PUBLICATION

AEP-67
(Edition 1)

ENGINEERING FOR
SYSTEM ASSURANCE
IN
NATO PROGRAMMES

AEP-67
EDITION 1

FEBRUARY 2010



Source: Walker (2005)

Figure 4-2 Supply Chain

Summary issues

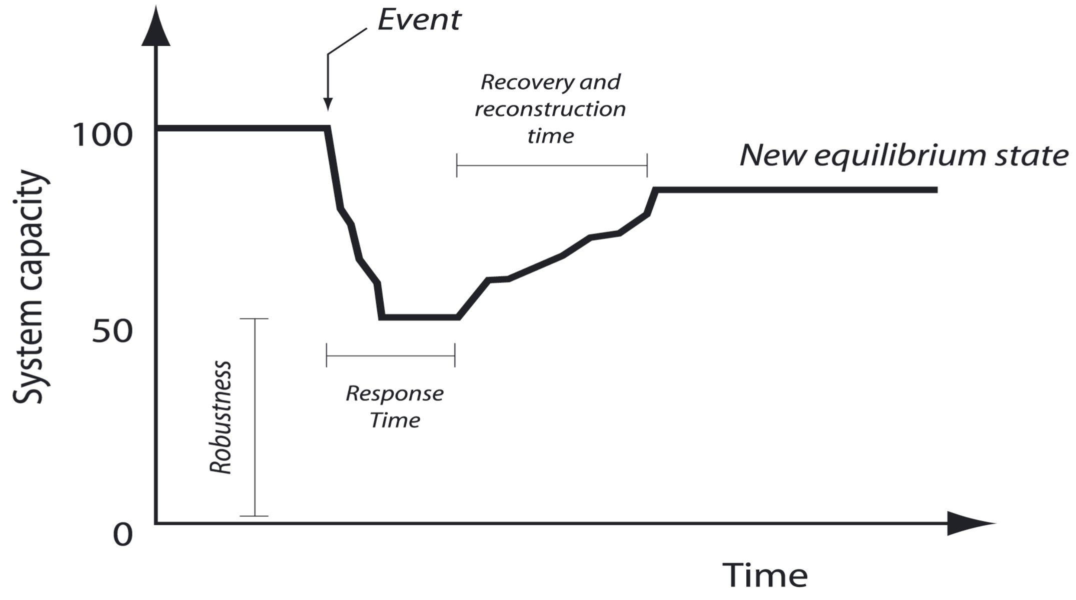
- identifying stakeholder assurance and communication needs
  - support for SLAs
  - resilience perspective
  - factor on threats
- information sharing and visibility
- gap between evaluated and deployed - claim architecture
- heterogeneous supply chain
- supply chain co-operation
- inevitability of probability
- dynamic cases
- composition

SCRM scope

- system assurance (SA) is *the justified confidence that the system functions as intended and is free of exploitable vulnerabilities, either intentionally or unintentionally designed or inserted as part of the system at any time during the life cycle.*
- What is the system? What is the environment? what are the threats?
- Who is making a decision about what, when?

- guidance recognises this is not possible and will need to have a case that balances risks
- *Inevitability of probability* for discussing risks and mitigations
    - cultural, model and data gap, link between measures/controls and what they achieve

Scoping issues - resilience viewpoint



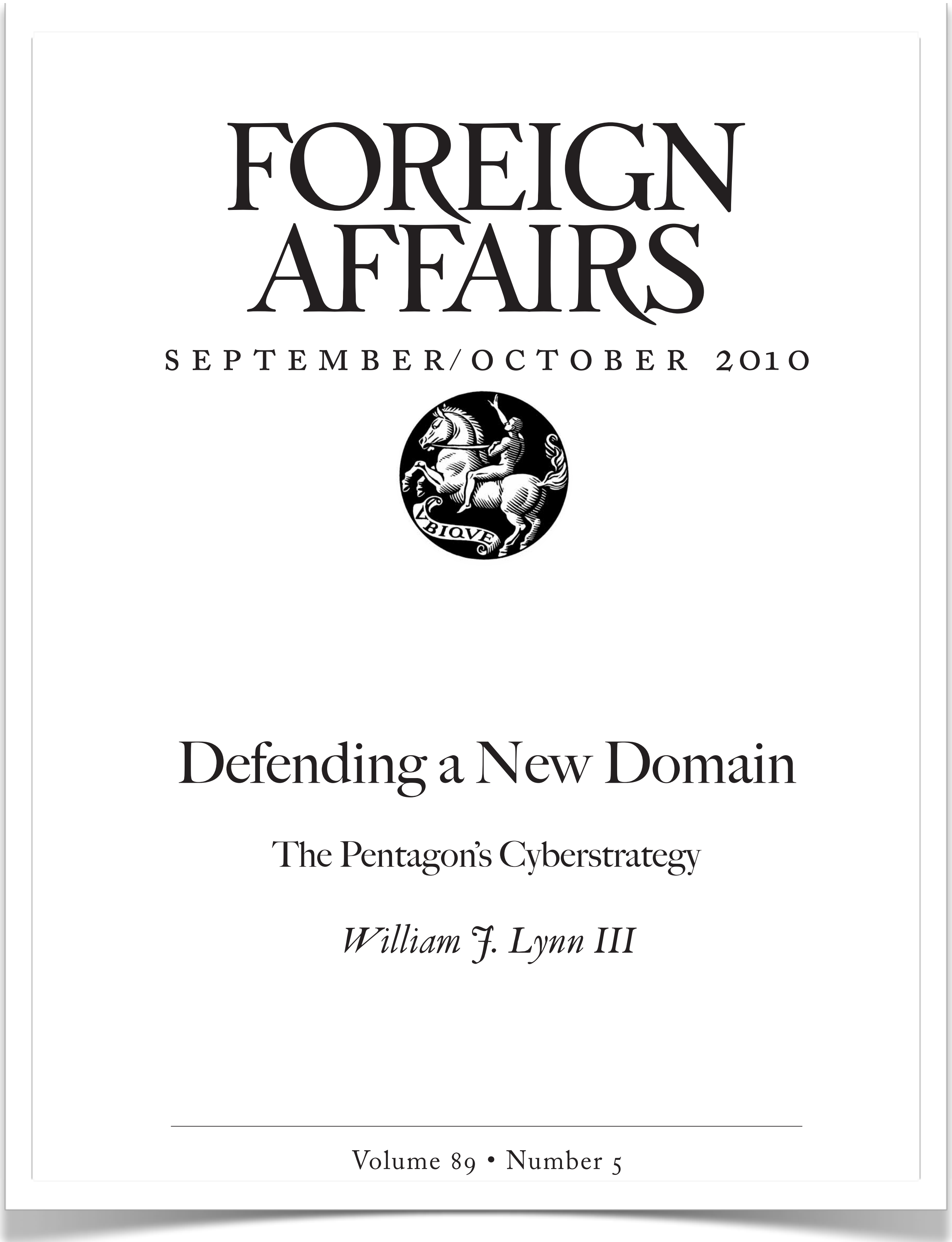
- *Type 1*: Resilience to design basis threats. This could be expressed in the usual terms of availability, robustness, etc. It could be bounded by credible worst case scenario.

- *Type 2*: Resilience to beyond design basis threats. This might be split into those known threats that are considered incredible or ignored for some reason and other threats that are unknowns

- Attacks on intangibles - these are also societal assets, not just CIP

Threat assumptions

- Defending a New Domain, US
- Cyber and IA strategy, UK
- Hadden Cave - evidence
- Hacker in the hardware - Scientific American



FOREIGN AFFAIRS

SEPTEMBER/OCTOBER 2010

Defending a New Domain

The Pentagon's Cyberstrategy

William J. Lynn III

Volume 89 • Number 5

Service assurance cases

- all hazards approach
- security needs to be socio-technical in scope (insiders, maloperation, social engineering)
- support for SLAs - justifying credibility
- SLAs do not transfer risk
- need to be in language of risk for trade offs and stakeholders (i.e. quantified)

Gap between evaluated and deployed

- strengthen cases with stronger arguments between deployed and evaluated systems
  - supply chain integrity, tamper proofness and other design measures
  - review arguments from non-interference, completeness of behaviour
  - review trade-offs with resourcefulness and adaptability

Supply chain co-operation

- need a technical and management approach
- *irony of lack of cooperation*. Reasonable to assume attackers have the supply chain code but suppliers might not provide this to partners or their users
- identify benefits and safeguards for suppliers
- provide technical justification for SLAs
- variety of assurance strategies - from wrappers to analysis
- alignment of incentives; economic

Heterogenous supply chain

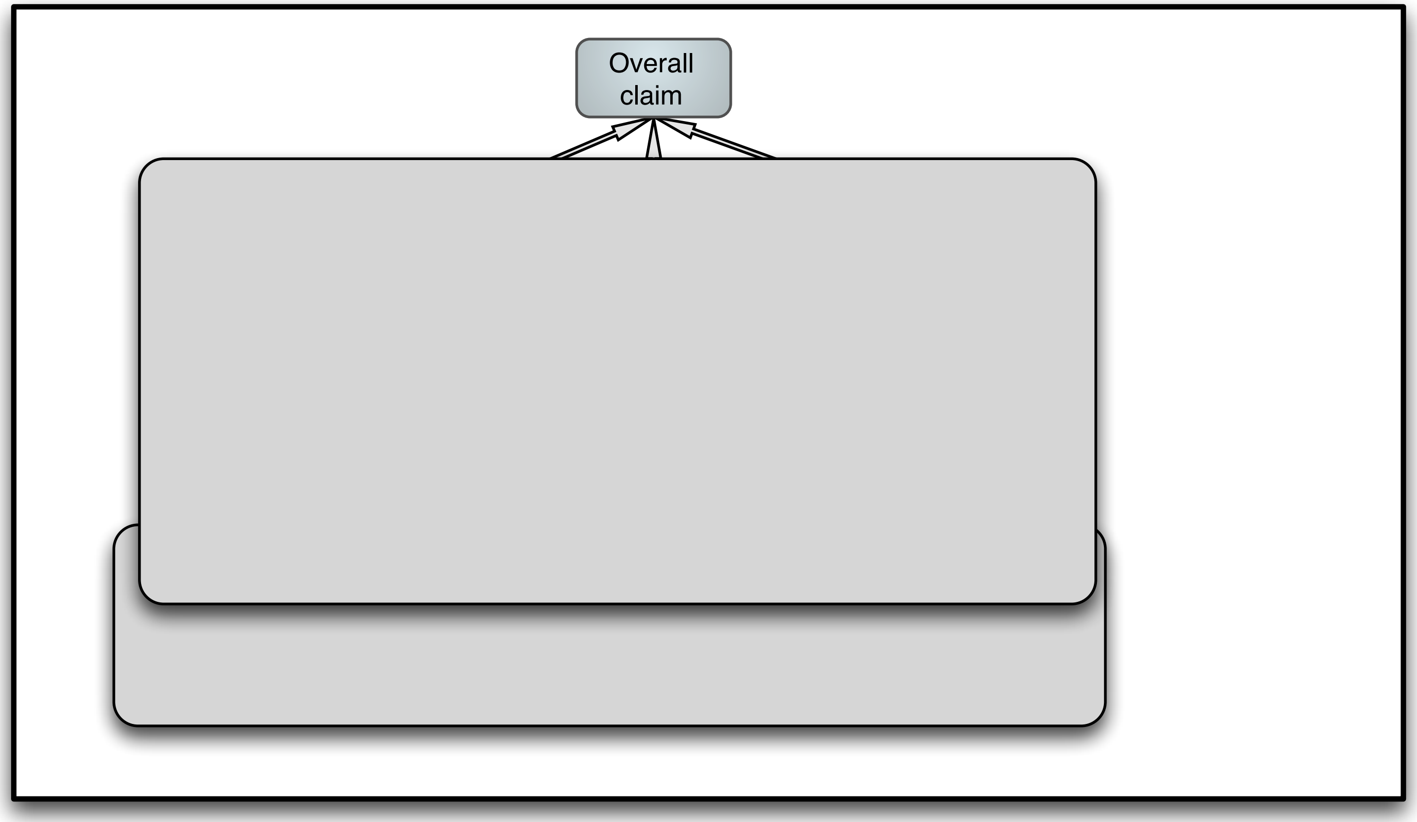
- heterogenous supply chain
  - flexibility in response and justification
  - range of strategies - openness, closure
  - trusted and uncertain sources
  - variety of threat assumptions
- interaction between assurance and system architecture
- even in safety need
  - security informed safety

Information sharing

- cases could provide systematic approach to
- assess confidence obtained by revealing/hiding parts of a case
- and the role of the meta-case

Top level claim

- reveal a partial claim that is adequate for the service of interest



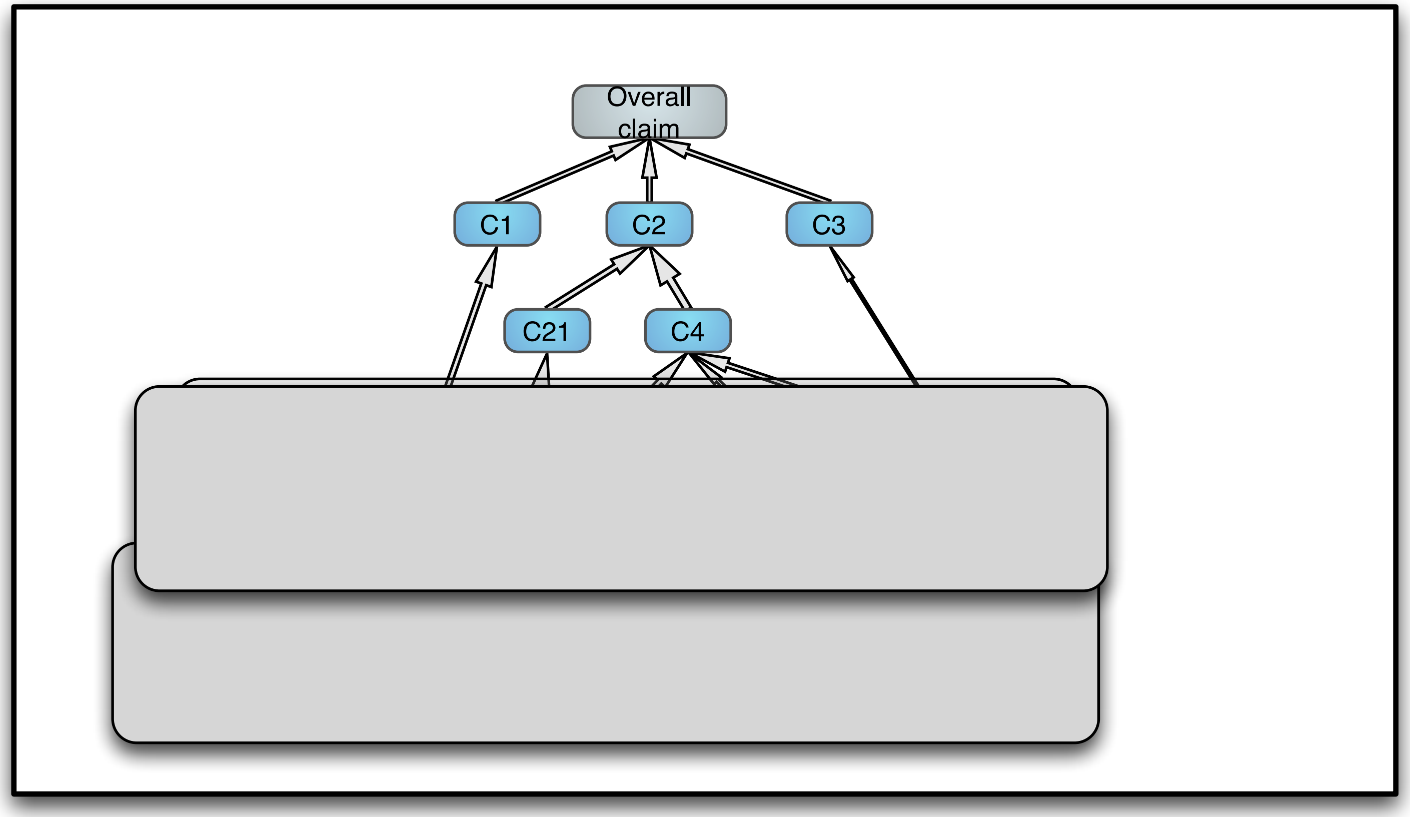
CSR Building confidence in a computerised world

www.csr.city.ac.uk

Adelard

Argument approach

- reveal overall thrust of the case

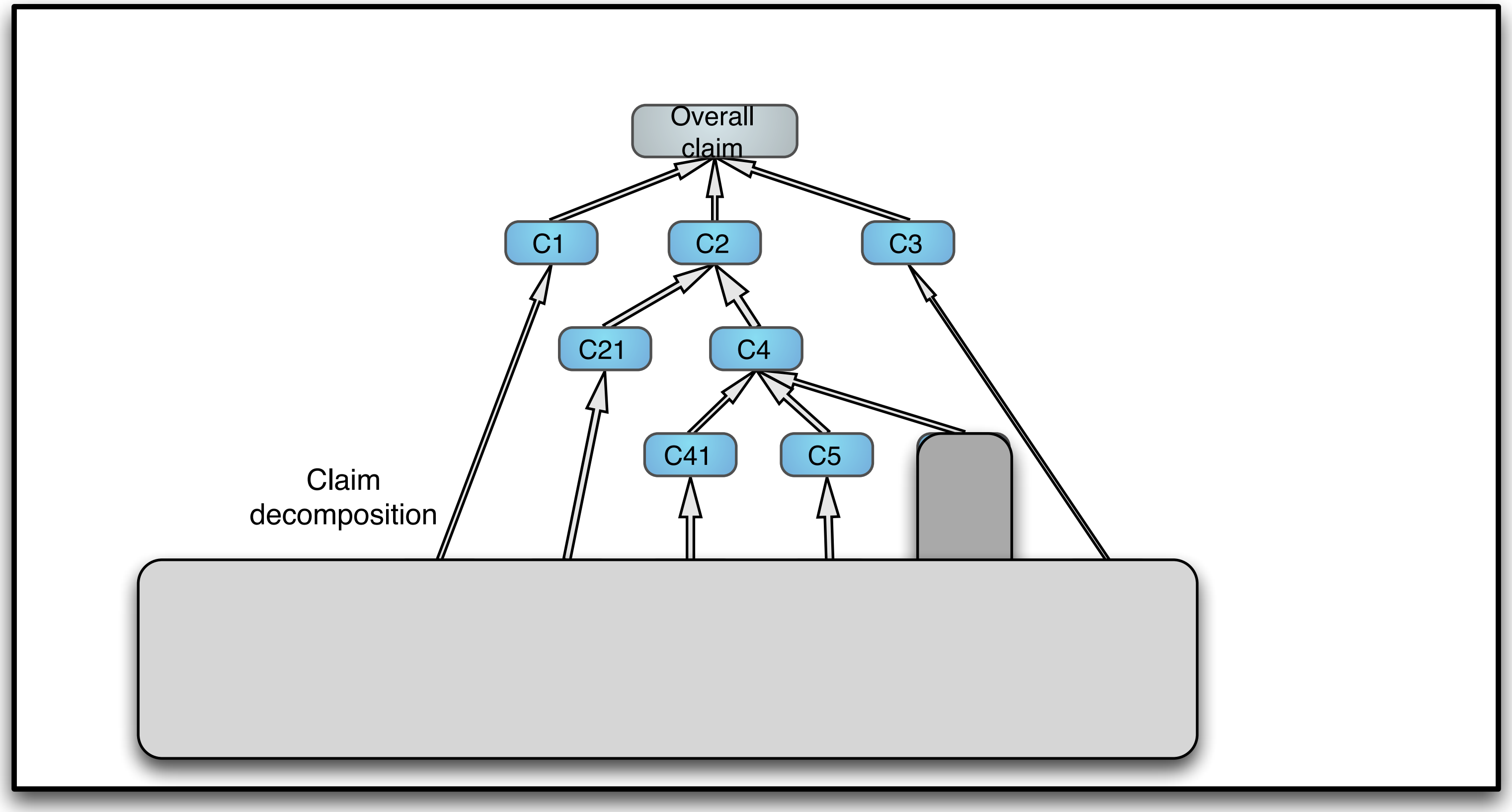


CSR Building confidence in a computerised world

www.csr.city.ac.uk

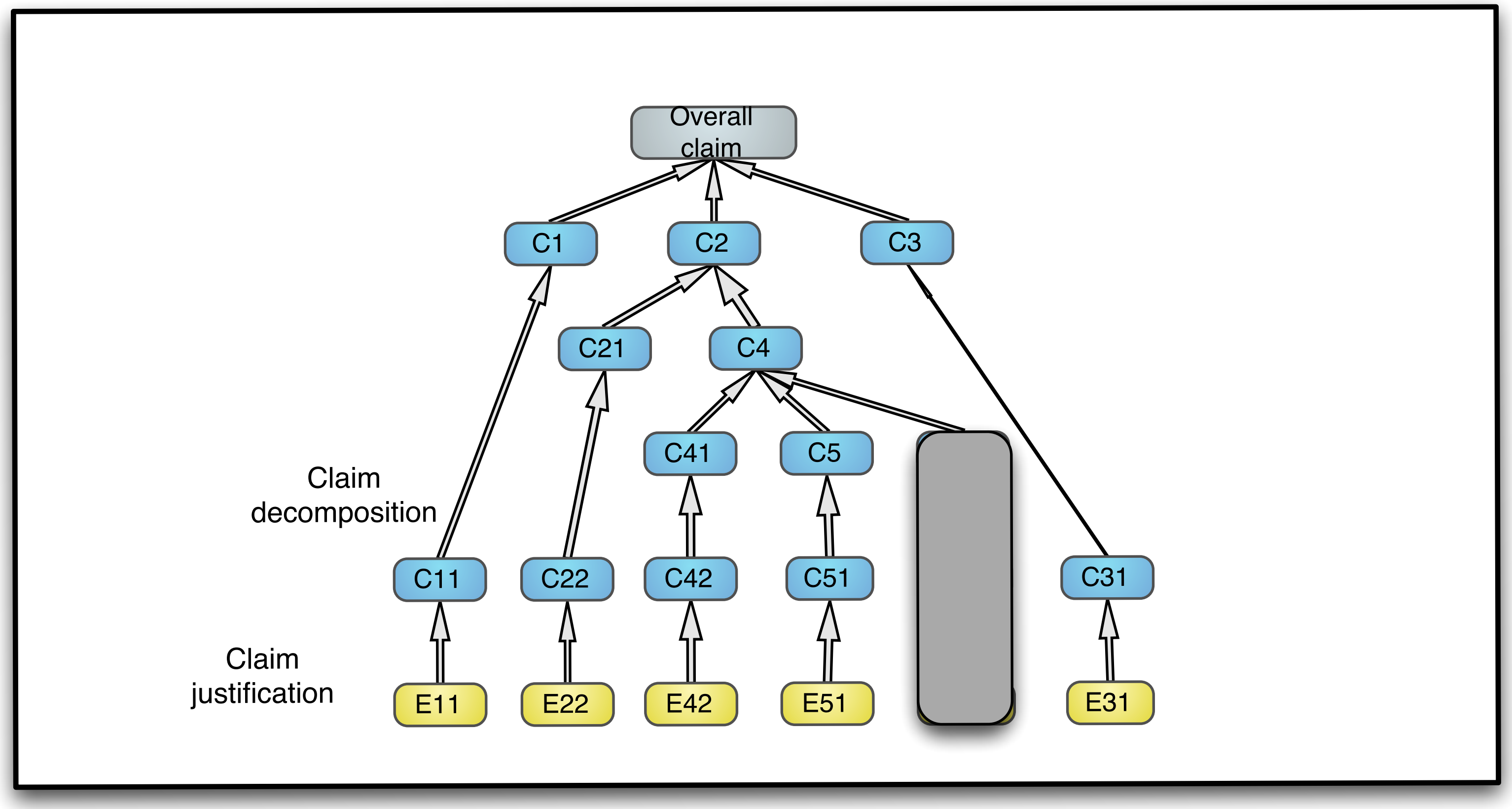
Adelard

- and some more details, but hiding capabilities



Evidence visibility

- graded evidence access from existence to details
- some evidence of how done will be graded in sensitivity

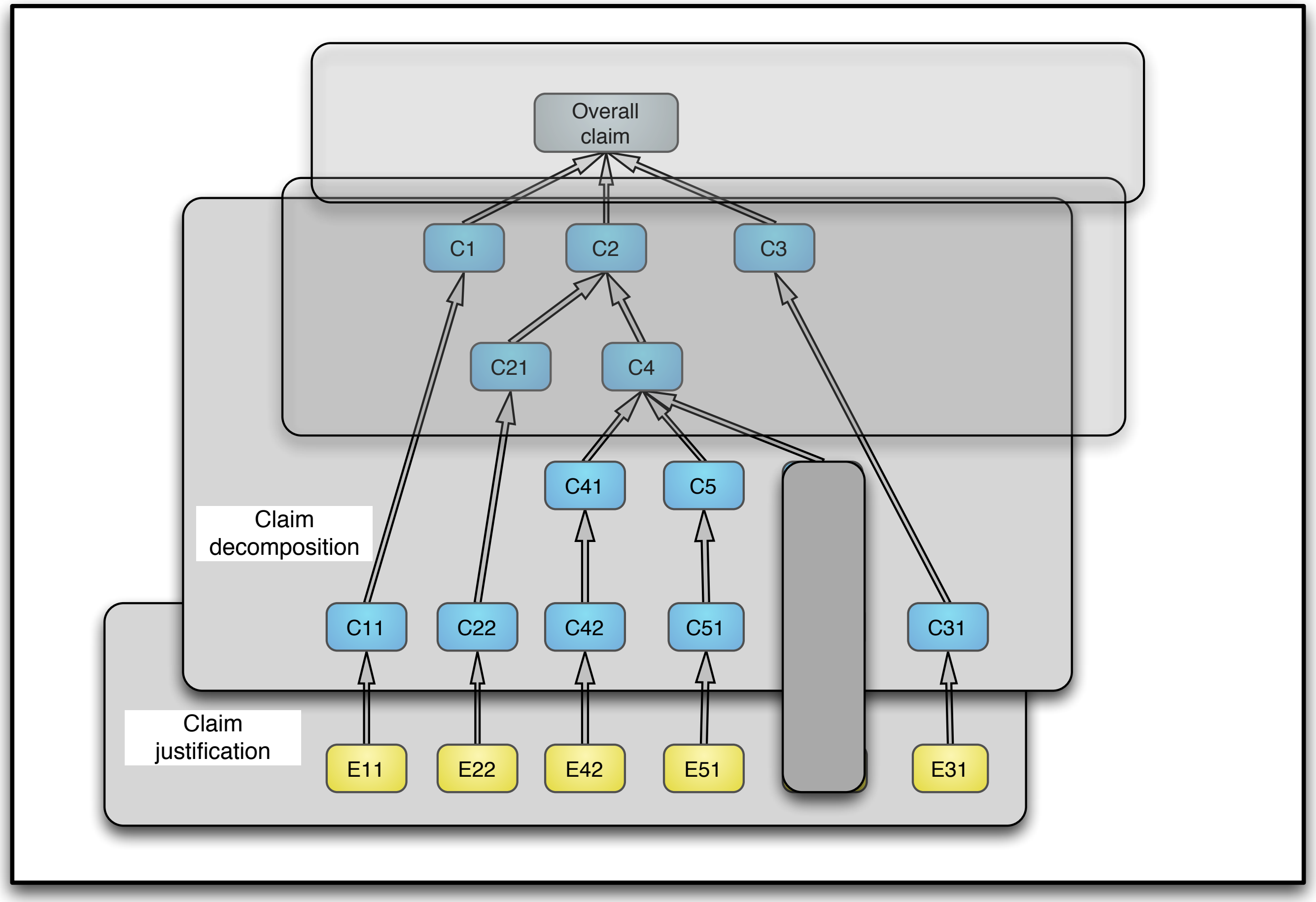


CSR Building confidence in a computerised world
www.csr.city.ac.uk

Adelard

Systematic approach to information sharing

- top down approach coupled with meta-case (why should I trust the case..)
- bottom up on what is willing to reveal
- principled approach to negotiating with supply chain



CSR Building confidence in a computerised world

www.csr.city.ac.uk

Adelard

Dynamic cases

- claim structure more static;
  - includes claims about ability to update and respond
  - as pattern for a range of scenarios
    - adjust, update, select
    - assets change
    - need to make rely assumptions clearer (e.g. positive behaviours)
  - pattern for different parts of resilience curve
    - normal levels of threat and response
    - incident response
    - heightened threat levels

Summary issues

- identifying stakeholder assurance and communication needs
    - support for SLAs, resilience perspective, factor on threats
- information sharing and visibility
- gap between evaluated and deployed - claim architecture
- heterogeneous supply chain
- supply chain co-operation
- inevitability of probability
- dynamic cases
- composition
- engine not a camera

CSR Building confidence in a computerised world

www.csr.city.ac.uk

Adelard

The promise of assurance cases

- Innovation in systems and assurance technologies
  - Can see how to incorporate new evidence
  - Cope with change, principled non-compliance
- Innovation in justification arguments and evidence
- Expose lack of validation of standards, gaps in our knowledge
- Focus of assessment and challenge
  - Need supporting safety case process and meta-case
- Clarity in the basis for regulation and licensing
  - See shortcomings of present approaches
- Improved communication with stakeholders
- Improved knowledge management
- Scalable
  - From smart components to complex systems
- Multi-attribute
  - Dependability, safety , security

Threat of assurance cases

- Apply safety analysis to cases themselves to understand risks and mitigations
  - Systematically analyse the failure modes for safety cases, using a HAZOPS style technique
  - Rejecting satisfactory cases, accepting inadequate cases
- Expose lack of validation of standards, gaps in our knowledge
- Competencies and skills and deployment risks
  - need for more methodology, examples
- Negatives to avoid
  - outsourced, commoditised, lack of controlling mind
  - just another report - value marginalised, a cost
  - complex, unclear, inappropriate cases

Maturity indicators

- ASCE statistics

- 250 organisations in 15 countries, many 1,000s users

  Key users:

  BAE SYSTEMS, QinetiQ, Boeing, Lockheed Martin, Raytheon, Thales, Westland, MBDA, General Dynamics, Northropp Grumann, AugustaWestland, Selex, Atkins, Quintec, Logica CMG, HVR, AWE

  Bosch, TRW, Moore Industries, Mira, Entec

  British Energy, BNFL, SKI, Framatome, AVN

  CAA, NATS, IAA, Eurocontrol, Indra, Advantage, CSE, Ebeni, Helios, Weston Aerospace

  Mitre Corp, FDA, NASA, Elekta Oncology, Cardinal Health, Medtronic

  Frazer Nash, Strachan and Henshaw, SSMG, NNC, ERA, Praxis

  Westinghouse, Ansaldo, Thales Rail, Network Rail

  MoD: Tornado, Harrier, Chinook, Jaguar, Puma Gazelle, JSF, Sea King, Merlin, ARC, U/water weapons, Helicopter Engines, ALM, PGB, Eurofighter/Typhoon, SUAV(E), Sub IPT, HMNBs Clyde & Portsmouth, Astute, TA, Bowman, DOSG, NW IPT, SSMO, LSSO, ARC, GBAD

- OMG standardisation

- International interest - global

- ISO 50126, Nato

- ... but need

Next developments

- In response to recent accidents, professional responsibilities
- Aim to publish a revised Adelard Safety and Assurance Case Methodology
    - Solve IPR and confidentiality issues with sponsors of the work
    - Establish other sponsors and internal investment
    - Confirm business model
        - Provide as service to the community
        - Sell tools and services
- Develop channels for learning from experience
    - Improvement, research, validation
    - Education and competency initiative
- Extend to CIIP and SCRM

Conclusions

- Reviewed assurance case concept of claims, arguments, evidence - CAE
- Major strategies for architecting claim structures
- Mappings between techniques and evidence
- Technical approach for dynamic and static analyses
- Supply chain experience from nuclear industry and financial services
- Extending notion into resilience and assurance cases and SCRM
- Aspiration to consolidate, publish and give away

Acknowledgments

- Colleagues in CSR and Adelard, particularly Peter Bishop, George Cleland, Lukasz Cyra, Sofia Guerra, Dan Sheridan, Bev Littlewood, Andrey Povyakalo, Lorenzo Strigini and others